

Open Research Online

The Open University's repository of research publications and other research outputs

Visualisation and manipulation tools for Modal logic

Thesis

How to cite:

Oliver, Martin John (1998). Visualisation and manipulation tools for Modal logic. PhD thesis The Open University.

For guidance on citations see [FAQs](#).

© 1998 The Author



<https://creativecommons.org/licenses/by-nc-nd/4.0/>

Version: Version of Record

Link(s) to article on publisher's website:

<http://dx.doi.org/doi:10.21954/ou.ro.0000dc90>

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online's data [policy](#) on reuse of materials please consult the policies page.

oro.open.ac.uk

**PAGE
NUMBERING
AS
ORIGINAL**



UNRESTRICTED

Martin John Oliver, B.A. (Hons)

Visualisation and Manipulation Tools
for Modal Logic

Submitted for the Degree of
Doctor of Philosophy in
Educational Technology

December, 1997

OPEN UNIVERSITY
AUTHOR'S NUMBER: M7193575

DATE OF SUBMISSION: 18 DECEMBER 1997
DATE OF AWARD: 16 JULY 1998

19 MAY 1999

RESEARCH DEGREES CENTRE

LIBRARY AUTHORISATION FORM

Please return this form to the The Research Degrees Centre with the two bound copies of your thesis to be deposited with the University Library.

All students should complete Part 1. Part 2 only applies to PhD students.

Student: MARTIN OLIVER PI: M7193575 ~~2255~~

Degree: PhD

Thesis title: VISUALISATION & MANIPULATION TOOLS
FOR MODAL LOGIC

Part 1 Open University Library Authorisation [to be completed by all students]

I confirm that I am willing for my thesis to be made available to readers by the Open University Library, and that it may be photocopied, subject to the discretion of the Librarian.

Signed: [Signature] Date: 14th MAY, 1999

Part 2 British Library Authorisation [to be completed by PhD students only]

If you want a copy of your PhD thesis to be available on loan to the British Library Thesis Service as and when it is requested, you must sign a British Library Doctoral Thesis Agreement Form. Please return it to the Research Degrees Centre with this form. The British Library will publicise the details of your thesis and may request a copy on loan from the University Library. Information on the presentation of the thesis is given in the Agreement Form.

Please note the British Library have requested that theses should be printed on one side only to enable them to produce a clear microfilm. The Open University Library sends the fully bound copy of theses to the British Library.

The University has agreed that your participation in the British Library Thesis Service should be voluntary. Please tick either (a) or (b) to indicate your intentions.

[a] ☒ I am willing for the Open University to loan the British Library a copy of my thesis.
A signed Agreement Form is attached.

[b] ☐ I do not wish the Open University to loan the British Library a copy of my thesis.

Signed: [Signature] Date: 14th MAY, 1999

Visualisation and Manipulation Tools for Modal Logic

Abstract

In this thesis, an investigation into how visualisation and manipulation tools can provide better support for learners of Modal logic is described. Problems associated with learning Modal logic are also researched.

Seven areas topics in Modal logic are investigated, as is the influence of domain-independent factors (e.g. motivation) on learning. Studies show that students find concepts such as Modal proofs and systems difficult to learn, whilst possible worlds and Modes are fairly straightforward. Areas such as reference, belief and accessibility relations fall between these extremes.

Two roles for representations in reasoning are identified: providing a concrete domain for students to reason about, and supporting the process of reasoning. Systems which make use of these complementary representations were found to be more effective for learners than either the syntactic or the diagrammatic representations traditionally used to teach Modal logic.

A review of software used to support students learning logic highlights two important features: the use of examples, and automation of routine tasks. A learning environment for Modal logic was designed which incorporated these. The environment was developed using an adapted version of Smalltalk's Model-View-Controller mechanism, and incorporates complementary representations, enhance by direct manipulation.

A further study investigates the added benefits of using this tool, as opposed to using the same representation but working with pen and paper. This confirms the importance of using 'concrete' content representations and minimising learners' cognitive load. Performance measures show that software users learnt more, had a deeper style of learning, and found the topics less abstract than their counterparts working with pen & paper.

This research shows that complementary representations are an effective way of supporting students studying Modal logic, and that visualisation and manipulation tools which incorporate these systems will provide additional benefits for learners.

Acknowledgements

I would like to thank Tim O'Shea and Pat Fung for their guidance, support and perseverance throughout this research, not to mention their reading and re-reading of innumerable drafts of this thesis. I would also like to thank Yibing Li for supervising my lengthy struggle with Smalltalk, and the ESRC for funding my studentship.

I am also indebted to all those who have commented on my work, papers and chapters, including Jane Barnard, Erica Morris, Eileen Scanlon, and Denise Whitelock. Their comments have been invaluable in getting this manuscript into an acceptable state.

Particular thanks are due to Cathy Lewin and Palitha Edirisingha, who made the sojourn in the office bearable, and whose camaraderie has helped me to survive the Ph.D. Shaaron Ainsworth also gets a special mention, as she not only proof-read the whole thesis, but was willing to take a rather naive young researcher under her wing and make sure I knew where to look for the skills and background information needed to make this story coherent.

I am also grateful to Grainne Conole, for giving me the time I needed to finish the thesis even though I was being paid to do something else entirely.

Penultimately, thanks to my family, who have always humoured my pretensions towards being educated, and to Kathy, whose love and guidance have shaped not just this research but pretty much the whole of my life. She has been a wonderful and much-needed distraction when work became oppressive, but equally a much-needed conscience whenever I found excuses not to finish the manuscript.

Finally, I'd like to thank God that this whole endeavour is finally over.

Contents

Chapter 1

Introduction	1
1.1 Introduction.....	1
1.2 The motivation for this research.....	2
1.3 Research questions.....	2
1.4 Methodology.....	3
1.5 The structure of this thesis.....	3
1.4 Summary.....	5

Chapter 2

Review of Previous Research.....	6
2.1 Introduction.....	6
2.2 Learning Modal logic.....	7
Key topics in Modal logic teaching texts.....	8
Summary.....	11
2.3 Logic teaching software.....	11
Proof Construction: JAPE	11
Tools for exploring applications of logic: MiraCalc.....	12
Visualisation tools: Venn.....	13
Learning Environments: Tarski's World.....	14
Summary.....	16
2.4 Graphical representations.....	16
Advantages and disadvantages	17
Systems of classifying graphical representations.....	18
The relationship between classification systems	22
Summary.....	22
2.5 Complementary ways of using graphical representations.....	22
Complementary representations for learning Modal logic.....	25
Summary.....	25
2.6 Conclusion.....	26

Chapter 3

Using Complementary Representations to Support

Learners of Modal Logic.....	27
3.1 Introduction.....	27
3.2 A new representational system for Modal logic.....	27
Identifying the need for a new system.....	27
An overview of the new system	28
3.3 A comparative study of the three representations.....	30
Methodology.....	31
Hypotheses.....	31
Subjects	32
Materials.....	32
Design.....	33
Procedure	33

3.4 Results: Pre- to post-test measures.....	34
The Distribution of Subjects	34
Use of Modal operators.....	35
Attempting operator reduction.....	36
Possible worlds	36
Reference.....	37
Belief and Modal operators.....	37
Modal systems	38
Modal proof construction.....	39
Near-Transfer.....	39
3.5 Results: Post-test measures	40
Time.....	40
Motivation	40
Representational preferences.....	41
3.6 Results: Qualitative data	42
Not all topics in Modal logic are equally difficult.....	42
Examples help concept acquisition and comprehension.....	42
Exercises support knowledge acquisition.....	43
Abstract notation is intimidating.....	43
Symbols are difficult to interpret.....	44
Manipulation skills can be acquired without being understood	44
The diagrammatic condition is not pictorial.....	44
The learning environment representation is more 'real' than the others.....	45
Assessment favoured the syntactic and diagrammatic conditions	46
Summary	46
3.7 Results: Analysis of the case study interview.....	47
The Effect of Prior Knowledge	47
Low motivation and perceived difficulty form a vicious circle	47
Understanding Possible Worlds	47
Problems with Reference	48
Interpretation requires Context.....	49
Interpreting symbols.....	49
Interpreting negatives.....	50
Self-explanation and visualisation helped identify areas of conceptual difficulty.....	50
The benefits of 'ownership' of a representation.....	53
Summary	53
3.8 Discussion.....	54
Instructional condition.....	54
Expertise	55
Time.....	55
Motivation	56
Gender.....	56
Domain related results.....	56
The effectiveness of the learning environment representation	57
Summary	58
3.9 A cognitive account of the performance differences between instructional conditions	59
Viscosity	61
Premature Commitment.....	61
Abstraction level	61
Summary	62
3.10 Conclusions.....	62

Chapter 4

The Design and Implementation of MoLE.....	64
4.1 Introduction.....	64
4.2 Design Aims.....	64
Interactivity.....	64
Constraining errors	65
Offering explanatory examples.....	65
Automation of rote tasks.....	65
General considerations.....	65
4.3 Adapting the representation for use in the software.....	66
The learning environment representation.....	66
MoLE's representation	67
Summary.....	69
4.4 Software Design.....	69
Design requirements.....	69
The Model.....	71
The View.....	71
The Controller.....	71
Object relationships and dependencies	72
4.5 Implementation.....	72
A generic VisualWorks object.....	72
Implementation of the Model.....	74
Implementation of the Views.....	77
Implementation of the Controllers.....	78
Part 1: 'Controller' sub-classes.....	78
Part 2: 'ApplicationModel' sub-classes.....	80
Design Refinements	84
Object relationships and dependencies	84
4.6 Conclusions.....	85

Chapter 5

An Overview and Formative Evaluation of MoLE.....	86
5.1 Introduction.....	86
5.2 An overview of the use of MoLE.....	86
The problem being investigated	86
Establishing the initial conditions.....	87
Constructing the diagram.....	90
Evaluating Claims	92
Summary.....	97
5.3 A Formative Evaluation of MoLE.....	97
Design.....	97
5.4 Results.....	98
Design issues	98
Implementation issues	99
Interface issues.....	99
Platform-related issues	100
Summary.....	100
5.5 The Redesign of MoLE.....	100
Provision of instructions.....	100
Alterations to the class GridWorld	100
Alterations to the class GridWorldInterface	100
Alterations to the class UniverseController.....	101
Alterations to the class UniverseView.....	101
Alterations to the class UniverseInterface.....	101
5.6 Conclusions.....	102

Chapter 6

An Evaluative Study of MoLE104

- 6.1 Introduction104
- 6.2 The Study.....105
 - Methodology105
 - Hypotheses105
 - Subjects.....106
 - Materials.....106
 - Design108
 - Procedure.....109
- 6.3 Results: Pre- to Post-test measures.....109
 - Distribution of subjects.....109
 - Accessibility relations.....109
 - Modal systems.....110
 - Proof construction110
 - Use of examples and/or counter-examples110
 - Application of systems (operator reduction).....111
 - Near transfer.....112
 - Summary113
- 6.4 Results: Post-test measures113
 - Subjects' understanding of Modal systems113
 - Subjects' understanding of proof construction114
 - Motivation114
 - Comments about the instructional conditions115
 - Number of exercises attempted116
 - Other Effects116
 - Summary117
- 6.5 Results: Qualitative data118
 - Assessment of the representation118
 - Problems with Modal systems.....119
 - Problems with Proof Construction.....119
 - Further issues.....120
 - Summary125
- 6.6 Discussion.....125
 - Instructional condition.....125
 - Expertise126
 - Gender.....127
 - Domain-specific problems.....127
 - Other factors influencing learning.....127
- 6.7 Conclusions129

Chapter 7

Conclusions.....131

- 7.1 Introduction131
- 7.2 A review of the research questions.....131
 - Topic-specific issues.....131
 - Representational questions.....132
 - Software issues133
 - Summary134
- 7.3 Contributions to knowledge.....134
 - Modal logic.....134
 - Learning with graphical representations.....136
 - Software design137
 - Cognitive Science.....138
 - Specific contributions138

7.4	Limitations of the current research.....	138
	Combining graphical representations.....	139
	Software implementation.....	139
	Empirical studies.....	139
7.5	Further Research.....	140
	Further investigation of topics in Modal logic.....	140
	Further investigation of complementary representations.....	140
	The transfer of representations to a new domain.....	140
	Comparison with other pairs of representations.....	141
	Further development of MoLE.....	141
	Adapting the software to other domains.....	141
	Investigation of the effect of users' perceptions on cognitive dimensions.....	142
7.6	Summary.....	142
References		144

Appendix 1: Study material, first study.....	153
Glossary of Symbols.....	153
Truth Tables for First Order Logic	153
Summary of Rules from First-Order Logic	154
Pre-Test.....	155
Intervention material 1.....	159
Intervention material 2.....	163
Intervention material 3.....	170
Post-test: Section A.....	180
Appendix 2: Quantitative data, first study.....	184
Appendix 3: Qualitative data, first study.....	192
Syntactic condition.....	192
Diagrammatic condition.....	197
Learning Environment condition	202
Appendix 4: Interview transcript for subject 38, first study.....	209
Appendix 5: Workscratchings, first study.....	227
Diagrammatic condition.....	227
Learning Environment condition	230
Appendix 6: MoLE code.....	233
Appendix 7: Observational logs, second study	281
Appendix 8: Study material, third study	285
Pre-Test	285
Instructional material	289
Exercises	302
Post-Test.....	308
Appendix 9: Quantitative data, third study	315
Appendix 10: Qualitative data, third study	321
Pen & Paper: Male novices.....	321
Pen & Paper: Male experts.....	322
Pen & Paper: Female novices.....	323
Pen & Paper: Female experts.....	323
Computer: Male novices.....	324
Computer: Male experts.....	324
Computer: Female novices.....	325
Computer: Female experts.....	325

Figures

Chapter 2

Review of Previous Research.....	6
Figure 2.1	
a syntactic representation of an accessibility relation (denoted as R).....	9
Figure 2.2	
a diagrammatic representation of an accessibility relation (denoted by	
arrows)	9
Figure 2.3	
a syntactic representation of a Modal relationship.....	10
Figure 2.4	
a diagrammatic representation of a Modal relationship.....	10
Figure 2.5	
The software tool JAPE.....	12
Figure 2.6	
The program MiraCalc	13
Figure 2.7	
The program Venn	14
Figure 2.8	
a screenshot from the program Tarski's World	15
Figure 2.9	
A screenshot from the program Hyperproof	16
Figure 2.10	
Comparison by fidelity.....	20

Chapter 3

Using Complementary Representations to Support

Learners of Modal Logic.....	27
Figure 3.1	
A grid world with one circle, two walls, and three letters.....	28
Figure 3.2	
An example of a relationship between possible worlds.....	29
Figure 3.3	
a transitive relationship, expressed syntactically.....	29
Figure 3.4	
a transitive relationship, expressed diagrammatically.....	29
Figure 3.5	
a transitive relationship, combining content and process representations	30
Figure 3.6	
Comparison of average scores on the operator reduction question.....	36
Figure 3.7	
A comparison of the responses to the post-test reference question.	37
Figure 3.8	
A comparison of the average number of Modal systems understood	39
Figure 3.9	
Expressed preferences for different representational styles.	42
Figure 3.10	
An example work-scratching showing the grid worlds created by the	
subject.	52

Figure 3.11	
A work-scratching showing relational information only	57
Figure 3.12	
A work-scratching showing 'content' information only	58
Figure 3.13	
Work-scratching showing a decomposition of a statement.....	59

Chapter 4

The Design and Implementation of MoLE.....	6 4
Figure 4.1	
The learning environment representation, as described in section 3.2.....	67
Figure 4.2	
A grid world using the revised system.....	69
Figure 4.3	
a Modal diagram using the revised system.....	69
Figure 4.4	
An example of an initial grid world.....	70
Figure 4.5	
The modal diagram created by allowing spheres to move once	70
Figure 4.6	
The relationship between the objects used in MoLE.....	72
Figure 4.7	
An example of the final GridView.....	78
Figure 4.8	
An example of the final UniverseView.....	78
Figure 4.9	
The hand icon	79
Figure 4.10	
An example of a window supported by GridWorldInterface	81
Figure 4.11	
An example of a window supported by UniverseInterface.....	83
Figure 4.12	
The revised relationship between the objects used in MoLE.....	84

Chapter 5

An Overview and Formative Evaluation of MoLE.....	8 6
Figure 5.1	
The initial Universe window.....	87
Figure 5.2	
Using the "set up initial conditions" dialogue to set the grid size.....	88
Figure 5.3	
The view after placing the first world.....	88
Figure 5.4	
The grid world window	89
Figure 5.5	
The grid world window, with pieces placed.....	89
Figure 5.6	
Using the edit menu to create a new Modal diagram	90
Figure 5.7	
A Modal diagram in which pieces have moved once.....	91
Figure 5.8	
The resulting Modal diagram	92
Figure 5.9	
Selecting the world in which the claim is to be evaluated	93
Figure 5.10	
Building the claim.....	93

Figure 5.11	
The dialogue window for selecting which pieces to test.....	94
Figure 5.12	
Entering the choice of pieces	94
Figure 5.13	
The truth value of the claim.....	95
Figure 5.14	
Altering the Modal system	95
Figure 5.15	
MoLE offers the users the chance to inspect a counter-example.....	96
Figure 5.16	
A counter-example to the claim, "it must be the case that piece 1 is above piece 3", in S4.....	97
Figure 5.17	
The additional cursors (cross-hair and 'ban') used by UniverseController	101
Figure 5.18	
An example of a window supported by UniverseInterface.....	102

Chapter 6

An Evaluative Study of MoLE.....	104
Figure 6.1	
A Modal diagram.....	107

Tables

Chapter 2

Review of Previous Research.....	6
Table 2.1	
A functional scheme for describing representations	19
Table 2.2	
Degrees of specificity	20
Table 2.3	
A comparison of representations for content and process representations.....	24

Chapter 3

Using Complementary Representations to Support

Learners of Modal Logic.....	27
Table 3.1	
The pre- and post-test questions used to assess the concepts under investigation	32
Table 3.2	
group comparison.....	34
Table 3.3	
means and variances for measures of performance	35
Table 3.4	
Means and variances for scores on operator reduction.....	35
Table 3.5	
Numbers of subjects who attempted the operator reduction question.....	36
Table 3.6	
Number of subjects who understood possible worlds.....	36
Table 3.7	
Means and variances for scores on the problem of reference.....	37
Table 3.8	
Number of subjects who understood how Modal logic and belief interact	38
Table 3.9	
Means and variances for the number of Modal systems understood	38
Table 3.10	
Number of subjects who understood possible worlds.....	39
Table 3.11	
responses to the post-test near-transfer question	39
Table 3.12	
Means and variances for time blocks taken	40
Table 3.13	
Means and variances ratings of enjoyment.....	40
Table 3.14	
Descriptive summary of conceptual problems.....	42

Chapter 6

An Evaluative Study of MoLE.....104

Table 6.1	
questions and concepts in the pre- and post-tests.....	108
Table 6.2	
Subject distribution by instructional condition, experience and gender	109
Table 6.3	
Number of correct explanations of accessibility relations	109
Table 6.4	
Means and variances of the number of systems correctly explained.....	110
Table 6.5	
Number of correctly constructed diagrams.....	110
Table 6.6	
Number of subjects able to use examples correctly	111
Table 6.7	
Means and variances of scores on the operator reduction system	111
Table 6.8	
Number of subjects who attempted concrete and abstract questions	112
Table 6.9	
Means and variances of scores for the types of reduction questions attempted	112
Table 6.10	
Means and variances of scores for concrete operator reduction questions	112
Table 6.11	
Number of subjects who attempted the near transfer question.....	113
Table 6.12	
Subjects' self-assessed understanding of systems T, S4 and S5	113
Table 6.13	
Subjects' assessment of problems constructing proofs.....	114
Table 6.14	
Subjects' accuracy at assessing whether they had accurately constructed a proof.....	114
Table 6.15	
Means and variances of subjects' enjoyment.....	115
Table 6.16	
Means and variances of subjects' perceived improvement.....	115
Table 6.17	
subjects' opinions about the pros and cons of their instructional condition, and the total number of subjects in each of the categories, for comparison.....	115
Table 6.18	
Means and variances of the number of exercises attempted	116
Table 6.19	
Means and variances for the re-scored measure of ability to apply Modal operators to cancellation problems	116
Table 6.20	
Means and variances of scores for enjoyment, by condition and gender.....	117
Table 6.21	
Means and variances for the re-scored measure of ability to apply Modal operators to cancellation problems	117
Table 6.22	
Means and variances of scores for confidence, by site.....	117
Table 6.23	
Subjects' responses to questions about blocks-worlds, the number of exercises, and their preference for instructional condition.....	118
Table 6.24	
Number of subjects expressing concern at time pressures.....	121

Chapter 1

Introduction

1.1 Introduction

In this thesis, an investigation of the ways in which software can use graphical representations to support students learning formal reasoning is described.

The domain chosen for this investigation is Modal logic. This is an extension of first-order logic which introduces the concept of necessary and possible truths (things which must and things which could be true). To understand and apply these ideas formally, students have to master supporting concepts such as possible worlds and Modal systems. They also need to acquire skills such as the ability to construct Modal proofs. Since little is known about the problems associated with learning Modal logic, a subsidiary aim of this thesis is to discover which topics cause learners difficulty, and why.

Visualisation and manipulation tools must, by definition, use representations to convey information. Investigating how such tools can do this most effectively involves understanding the ways in which external representations are used to present and explore concepts. Research suggests that the way in which external representations are used in formal reasoning has a considerable impact on what is learnt, and has demonstrated the importance of selecting appropriate representations for a given task. Because of this, a clear specification is required of the intended uses of representations, together with an explanation of how they will satisfy the various needs of learners.

In order to arrive at this specification, a number of classification systems for representations are considered. This leads to the proposal of a simple distinction between the roles representations play during formal reasoning. Representations can be used in two ways: to provide content or to support the reasoning process. These roles are argued to be complementary. Based on this distinction, a new way of representing Modal logic has been developed. This combines the blocks-world metaphor of the programme Tarski's World (Barwise & Etchemendy, 1992) with the tree diagrams

traditionally used to teach Modal logic. Using blocks-worlds in this way raises a further research question: whether representations shown to be effective in one domain can be successfully re-used elsewhere.

The role of software tools in supporting students learning formal reasoning is also investigated. This process begins with a review of existing software tools, which has resulted in the identification of four key types of software: proof construction tools, tools for exploring applications of logic, visualisation tools, and learning environments. A comparison of these identifies two main design themes: the automation of rote tasks, in order to promote exploration of the domain and reduce unnecessary cognitive effort, and the provision of examples, to tackle issues of perceived irrelevancy of the topic and the lack of motivation common amongst students learning formal reasoning.

These features, together with the new complementary representation derived for Modal logic, lead to the design of a new software tool. An evaluation of the Modal Learning Environment (MoLE) allows an investigation of the ways in which visualisation and manipulation tools support learners, and in particular, whether the enhancement of the representation by direct manipulation and automation of routine tasks provides additional benefits for learners.

1.2 The motivation for this research

Learning any formal topic can prove difficult for students. Few topics illustrate these problems quite so well as logic, which epitomises abstract formal reasoning. The abstract way these subjects are taught traditionally makes them extremely hard to understand, leads to a perception that the topic has no 'real-world' relevance, and to poor motivation. Personal experience suggests that for many students, "either you get it or you don't"; the topic is either transparent or too dense to comprehend, with little or no middle ground.

Graphical representations offer a promising way to mediate these problems. Their use in related formal domains suggests that they can illustrate obscure concepts, clarify complex procedures, and improve motivation. They offer ways of supporting learning which are still to be fully understood, and which are currently under-used or neglected in domains such as Modal logic.

When these are combined with the support for learning offered by computers, the visualisation and manipulation tools which result should have much to offer students learning formal topics.

The increasing importance of formal methods, particularly in subjects such as computing, together with the use of Modal logic in courses as diverse as linguistics, philosophy and mathematics, makes the investigation described in this thesis particularly timely.

1.3 Research questions

In this thesis, the question of how software which incorporates graphical representations can be used to support students learning about formal reasoning is addressed. In order to answer this, the problem has been broken down into the following research questions:

- 1) What problems are associated with learning Modal logic?

- 2) What uses of graphical representations does current research suggest will effectively support formal reasoning?
- 3) Are these predictions supported empirically?
- 4) Can existing representations, such as the blocks-world metaphor of Tarski's World, be adapted for use in other domains?
- 5) What elements of existing programs can be used to support formal reasoning?
- 6) What software design will incorporate these elements with the representation best suited to supporting students learning Modal logic?
- 7) What benefits do learners get from using this kind of tool?
- 8) Are these benefits merely the result of the representation, or is there added value in incorporating the representation into a software tool in this way?

1.4 Methodology

A five-stage methodology was adopted in order to answer the research questions outlined above. The first stage involved reviewing existing literature on logic learning with graphical representations, in order to predict which systems are best able to support learners. Additionally, a review of existing software for students learning logic was used to identify features which are argued to be beneficial to learners.

The second stage involved empirically testing predictions about systems of graphical representations empirically. This also provided an opportunity to investigate research questions relating to the difficulties associated with learning Modal logic.

The results of this study were then combined with the findings of the software review to create a design for a new tool for students learning Modal logic. This specification, together with the tool's implementation, formed the third stage of the methodology. The fourth involved usability-testing for the tool, which lead to several design refinements.

Finally, the benefits of the new tool were assessed empirically. As with the first empirical study, this fifth stage allowed an investigation of domain-related issues.

1.5 The structure of this thesis

Having identified the research questions to be addressed, this section outlines the remainder of this thesis.

Chapter Two

This chapter discusses issues relating to learning Modal logic. It focuses on a review of current teaching texts and a summary of the topic-independent issues identified as influencing students' success at learning first-order logic. Additionally, software used to support logic learning is reviewed, leading to an identification of features argued to be effective for learners.

Previous research on graphical representations and their role in reasoning is then discussed. Six systems for classifying and comparing representations are presented, and a distinction between content and process diagrams is made.

Chapter Three

Chapter three derives a new system of representation for Modal logic, based on the recommendations for complementary graphical representation use identified in chapter two. This system combines a representation currently used to teach Modal logic (a 'process' diagram) with a representation from an existing piece of logic teaching software (a 'content' diagram), and is referred to as the learning environment style of representation.

The chapter then presents an empirical evaluation of the effectiveness of three styles of representation for learners of Modal logic. These are the syntactic and diagrammatic representations traditionally used to teach the subject, together with the new learning environment representation. In addition to investigating which style of representation is the most effective, the study examines measures of improvement across the three conditions in order to identify which Modal topics are difficult for learners, and why. Additionally, the study considers whether the blocks-world metaphor used to support the learning of first-order logic can be successfully adapted to Modal logic.

Chapter Four

The design and implementation of the Modal Learning Environment (MoLE) is outlined in chapter four. The learning environment representation, described in chapter three, is revised in the light of results of the comparative evaluation. This is then used as the basis of the tool.

After describing the pedagogic aims of MoLE, the software design and implementation are described.

Chapter Five

An overview of the software tool MoLE is presented. This involves a description of how the software can be used to investigate a typical problem from Modal logic.

Additionally, a formative study of the tool is described. This is used to identify problems and specify refinements for MoLE. The chapter closes by describing the revisions to MoLE implemented as a result of this study.

Chapter Six

An empirical evaluation of the software tool, MoLE, is described in chapter six. The performance of subjects using the tool is compared with that of subjects using the same representation but working with pen and paper.

The results of this evaluation are used to address questions about the benefits of this type of software tool for learners, and whether these benefits are solely the result of the representation it uses. They also provide further insight into how the blocks-world metaphor can be adapted to the domain of Modal logic.

Chapter seven concludes the thesis by revisiting the research questions, reviewing the previous chapters, and assessing the extent to which the evidence answers the questions which were posed. Other results identified during the research process are also presented, providing a summary of contributions to the field.

The limitations of the thesis are discussed, and short- and long-term research topics which build on the findings of this thesis are identified.

1.4 Summary

In order to investigate how visualisation and manipulation tools can be used to support students learning Modal logic, a five-stage methodology has been adapted. This involves a review of current literature, the design of a software tool, and a series of three empirical studies. These will be used to address the eight subsidiary questions which contribute to the research problem. In order to provide an overview of the thesis, its structure is described, and the aims and content of each chapter are specified.

Chapter 2

Review of Previous Research

2.1 Introduction

Logic is a topic usually taught at university level. It can be encountered in a variety of courses, as it is used in subjects such as mathematics, computer science, philosophy, and linguistics. Non-standard logics, including Modal logic, are not usually encountered until the second or third year of a degree. Typically, they will form part of a course for which familiarity with first-order logic is a pre-requisite.

The restrictions on Modal logic courses imply that this is a particularly difficult subject to learn. This chapter reviews research which elaborates on this assumption. A search of the literature pointed to a need for research into the difficulties associated with learning Modal logic. To provide a starting point for this, and to provide the necessary background for the research in this thesis, two related themes are reviewed: issues addressed by research on the process of learning first-order logic, and topics identified by textbooks as being central to Modal logic courses.

Several programs have been created to address the difficulties faced by students learning first-order logic. A selection of software that illustrates different approaches to supporting logic learning is presented, and common features, intended to support learning, are identified.

One issue which is identified in the Modal logic teaching texts and returned to in the logic teaching software is the way in which graphical representations can be used to support students learning logic. To develop a principled framework for using graphical representations in this type of software, the literature on learning with external representations is reviewed. This gives rise to a new classification system, which distinguishes between representations used to provide content for reasoning and those used to support the reasoning process. These are argued to be complementary.

2.2 Learning Modal logic

A search was carried out for literature discussing the issues associated with learning Modal logic. This showed a need for further research, since there appears to be a lack of empirical work on the problems associated with this topic. A pilot survey was conducted which asked lecturers to identify common difficulties and provide a list of the teaching texts they recommended (Oliver, 1995). Data from this were used to explore the problems that arise for learners of Modal logic, and to identify central topics of Modal logic courses.

It became apparent from the survey that students are often unmotivated by Modal logic, failing to see either its relevance or its applications. In general, they consider proof construction to be time consuming and “tricky”. Although these concerns had little to do with the topics that were covered in Modal logic courses, they closely resemble issues identified by studies of students learning first-order logic (e.g. Fung et al, 1993). Given the close relationship between the two topics, it seems likely that problems such as these, which are motivational or skill-based rather than being related to the domain, would be common to both Modal and first-order logic. Consequently, literature on learning first-order logic was reviewed in order to explore these issues further.

These two themes (issues involved in learning first order logic, and key topics in Modal logic) are presented in this section, in order to arrive at an understanding of the issues involved in learning Modal logic.

Problems associated with learning first-order logic

Logic learning poses many challenges for students. Learning abstract reasoning may require years of instruction, and may only lead to transfer of skills to closely related domains (van der Pal, 1996). Being an abstract topic, it seems to have little relation to meaningful, real world problems (Fung & O’Shea, 1992). This leads to motivational and conceptual difficulties, for example experiencing ‘shock’ when faced with ‘abstract things’, or an inability to use logical strategies such as modus tollens (e.g. Fung & O’Shea, 1994; Cheng et al, 1986).

A lack of prior training can further impede students’ performance on logical reasoning tasks (Fung & O’Shea, 1992). The most significant skills which were found to be lacking in this study were:

- familiarity with formal notation,
- the ability to break problems into manageable component sections,
- the skills required to manipulate formulae, and
- the ability to abstract general principles from particular cases.

There also seems to be a considerable gulf between natural and formal languages, for example, in terms of their expressive richness (Barwise & Etchemendy, 1990). This suggests that students who have already studied formal languages, and so have already overcome the gulf between everyday and formal reasoning, may find it easier to learn further formal topics. In support of this, previous experience with mathematics has been found to affect the performance of subjects learning first-order logic (Fung & O’Shea, 1994).

Topics presented as being conceptually difficult by teaching texts

Five teaching texts were identified by the pilot survey mentioned above (Oliver, 1995): *An Introduction to Modal Logic* (Hughes & Cresswell, 1968), *A Companion to Modal Logic* (Hughes & Cresswell, 1985), *An Introduction to Modal Logic* (Lemmon, 1977), *Logic for Computer Science* (Reeves & Clarke, 1990) and *Modal Logic* (Chellas, 1980). Of these, the most widely used was Hughes & Cresswell (1968).

These texts were reviewed in order to identify concepts considered to be important in Modal logic courses. Seven were identified. The first five of these are core concepts common to all Modal logic courses; the last two relate to common applications of the subject. These are: Modes, possible worlds, accessibility relations, Modal systems, Modal proofs, problems of reference, and issues about belief. These topics are outlined below.

1) Modes

Modes, the concept from which Modal logic takes its name, distinguish between whether something must be true, whether it might be true, and whether it must not be true.

For example, if all symbols are assumed to have their conventional meanings, it must be the case that $2+2=4$. Similarly, it might be the case that $2+x=4$, or that $2+x<4$, but it cannot be the case that $2+3=4$. Note that saying, " $2+x=4$ might be the case," (i.e. is possibly true) makes no claim as to whether or not the statement actually is true.

In formal notation, something which must be true is prefixed with the symbol, " \Box ", which denotes necessity. Something which might be the case is prefixed " \Diamond ", denoting possibility. Something which must not happen is prefixed " $\Box \neg$ ", or equivalently, " $\neg \Diamond$ ".

2) Possible Worlds

Introducing the concept of Possible worlds allows Modal logic to describe situations which, whilst actually untrue, could occur. The exact meaning of "could" here is context dependent. Possible worlds are sometimes called counterfactual situations.

An example of a possible world could be a situation identical to the real world in every regard, except that a name is spelt differently, or this text is printed on blue paper, or some other such variation. Formally, possible worlds are treated as a collection of propositions (each of which may be true or false) together with the proofs that can be derived from them.

3) Accessibility relations

Accessibility relations are introduced in order to explain how different possible worlds relate to each other. If the propositions that are true for one world can be transformed into those that are true for another, then the process of transformation is referred to as an "accessibility relation".

Accessibility relations are closely connected with the meaning of words such as "could" or "can". A situation "can" occur if the set of propositions for the original world can be transformed into the new set of propositions using whichever accessibility relation has been adopted. The choice of accessibility relation will be influenced by the topic and proof under investigation.

An example of an accessibility relation would be to wait five minutes. The truth values of propositions in the starting situation could alter in many different ways during this time, and each different outcome will result in a new possible world.

As with subsequent concepts, there are two common styles of representation used to denote accessibility relations. These are informationally equivalent (Larkin & Simon, 1987). Syntactically (e.g. in Lemmon, 1977; see Figure 2.1), accessibility relations are often denoted as a capital letter (typically R) which is placed between two symbols (each denoting a possible world).

If **A** has the form $\Diamond B$, then $\models_{U_0} A$ iff there exists t such
that uRt and $\models_t B$

Figure 2.1: a syntactic representation of an accessibility relation (denoted as R)

Diagrammatically (e.g. in Reeves & Clarke, 1990; see Figure 2.2), they can be represented as an arrow linking one possible world to another.

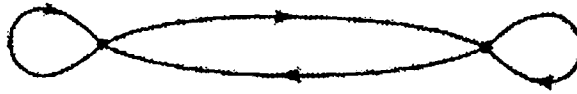


Figure 2.2: A diagrammatic representation of a reflexive, transitive accessibility relation (denoted by arrows)

4) Modal systems

In the same way that accessibility relations explain how worlds are connected, Modal systems are used to compare the patterns of connections that accessibility relations give rise to. They do this by grouping different accessibility relations together according to the properties they possess. This enables a fast assessment of whether a proof valid under one accessibility relation is also valid under another.

Modal systems are usually defined axiomatically, although the three most widely taught systems (called T, S4 and S5) can also be defined according to whether or not they are reflexive, transitive and symmetric. (T is reflexive, S4 is reflexive and transitive, and S5 is reflexive, transitive and symmetric.) These qualities are easily represented using diagrammatic conventions, for example by using arrows, double-headed arrows, and 'linked' arrows, in a way similar to that shown in Figure 2.2.

5) Modal proofs

Modal proofs involve demonstrating that a claim holds true over a particular set of possible worlds. The starting conditions and accessibility relation determine exactly what this set will comprise of.

Because Modal systems can be represented either syntactically or diagrammatically, there will be two informationally equivalent forms of every Modal proof. Because these are equivalent, any diagrammatically valid proof can also be proved sententially, and vice versa.

Proofs using the syntactic relationship are built up from lines such as the following (Figure 2.3):

$$((W R W') \ \& \ (W' R W'')) \rightarrow (W R W'')$$

Figure 2.3: a syntactic representation of a Modal relationship

This example states that if a possible world W is related to a world W' , and W' is in turn related to world W'' , then W is also related to W'' . In other words, the relationship R is transitive.

Proofs which adopt a diagrammatic style of representation (e.g. Hughes & Cresswell, 1968), are constructed with boxes (possible worlds), arrows (accessibility relations), formal sentences and truth values (denoted 0 or 1), as shown in Figure 2.4.

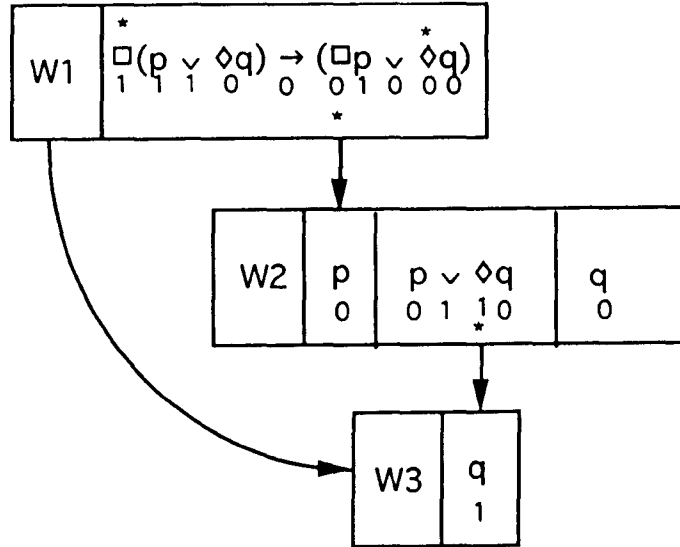


Figure 2.4: a diagrammatic representation of a Modal relationship

This example also represents transitivity. Here, the fact that $W1$ has an arrow to $W2$, and $W2$ has one to $W3$, means that an arrow can also be drawn from $W1$ to $W3$.

The rules for proof construction vary according to which Modal system is being used. For example, a proof which is S5 valid may not be T valid, since proofs in the different systems would have distinct structures, both syntactically and diagrammatically.

6) Reference

The concept of reference is the first of the two applications of Modal logic which were identified as key topics.

An important theme in Modal logic is that the name of an object (its 'sense') is not the same as the object it refers to (its 'reference'). For example, whilst we use the word 'computer' to refer to a machine which processes information, it would be easy to imagine a world in which 'computer' referred to a person who did arithmetic. If an accessibility relation allows the meaning of names to change, the nature of what can be proved alters too.

7) Belief

The second common application of Modal logic is to use it to model situations such as ethics or beliefs. In such areas, rules from first-order logic can fail, since it is possible for people to hold inconsistent beliefs. As an example, an agent might believe that $2+x=4$ irrespective of whether this was true, or even possible.

Although other topics also featured in the teaching texts identified by the survey, these seven form a common 'core' for most introductory courses to Modal logic.

Summary

A review of teaching texts has led to the identification of seven key areas which may pose particular difficulty to students learning Modal logic. In addition to these, several more general considerations may affect how students learn this topic. These problems include the abstract nature of logic, which can cause problems with interpretation and motivation, and that having prior formal experience provides the skills required to learn further formal languages more easily.

2.3 Logic teaching software

A wide variety of software intended to support logic learning has been produced. A broad selection of these programs have been reviewed already (e.g. Goldson et al, 1993; Fung et al, 1995). The programs in these reviews can be classified into four different types: proof construction tools, tools for exploring applications of logic, visualisation tools, and learning environments.

This section presents an example of each of these types of program, in order to characterise the types and provide an overview of current logic-teaching software.

Proof Construction: JAPE

Within the category of proof construction tools, there is a considerable difference in the extent to which the software automates the construction process. At one extreme there are programs similar to text editors dedicated to proof construction (e.g. Manna & Waldinger, 1985); at the other, there are programs which will generate completed proofs automatically (e.g. Dawson, 1991). The program JAPE (Just Another Proof Editor) falls between these two extremes (Bornat et al, 1996; Figure 2.5).

JAPE is described as being a 'proof calculator'; it makes correct steps in a proof under guidance from the user. As these steps are taken, the proof window is dynamically updated. Once completed, proofs can be stored as theorems for later use.

This balance between automation and user construction is intended to provide practice in proof construction whilst giving support and reducing 'book-work', which hinders the exploration of concepts (cf. Barwise & Etchemendy, 1990). Its intent is to provide practice of using proof construction rules, and so develop students' experience and confidence.

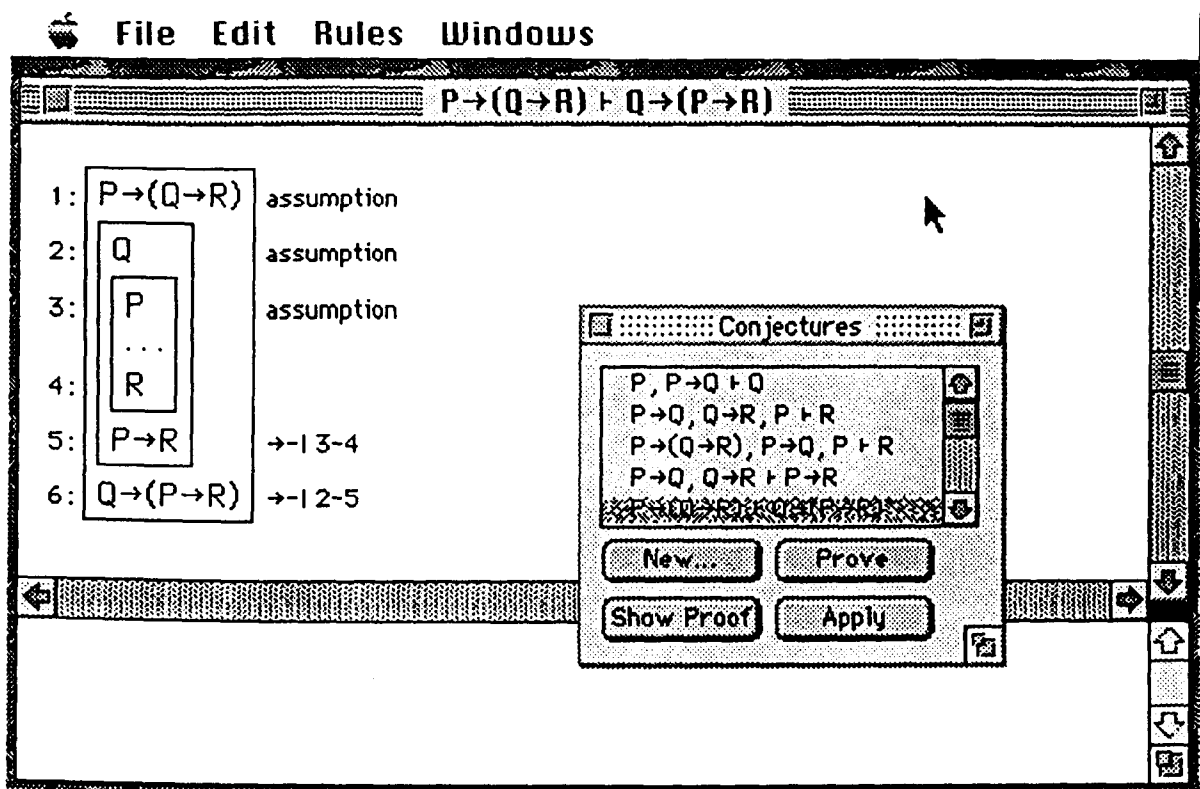


Figure 2.5: The software tool JAPE

Tools for exploring applications of logic: MiraCalc

MiraCalc also adopts a 'calculator' format. However, whereas JAPE concentrates on supporting proof construction, the aim of MiraCalc is to help students reason formally about functional programs (Goldson et al, 1994; Figure 2.6).

MiraCalc shares JAPE's aim of provide support for students by minimising typing, automatically reducing terms, and helping the user to explore the structure of programs. Where it differs is in its aim to bridge the gap between abstract concepts and putting those ideas into practice (Fung & O'Shea, 1994). It achieves this through the application of formal methods to a separate domain: in this case, functional programs. Combining formal training with concrete instances like this has been shown to be an effective way of teaching logic (Cheng et al., 1986).

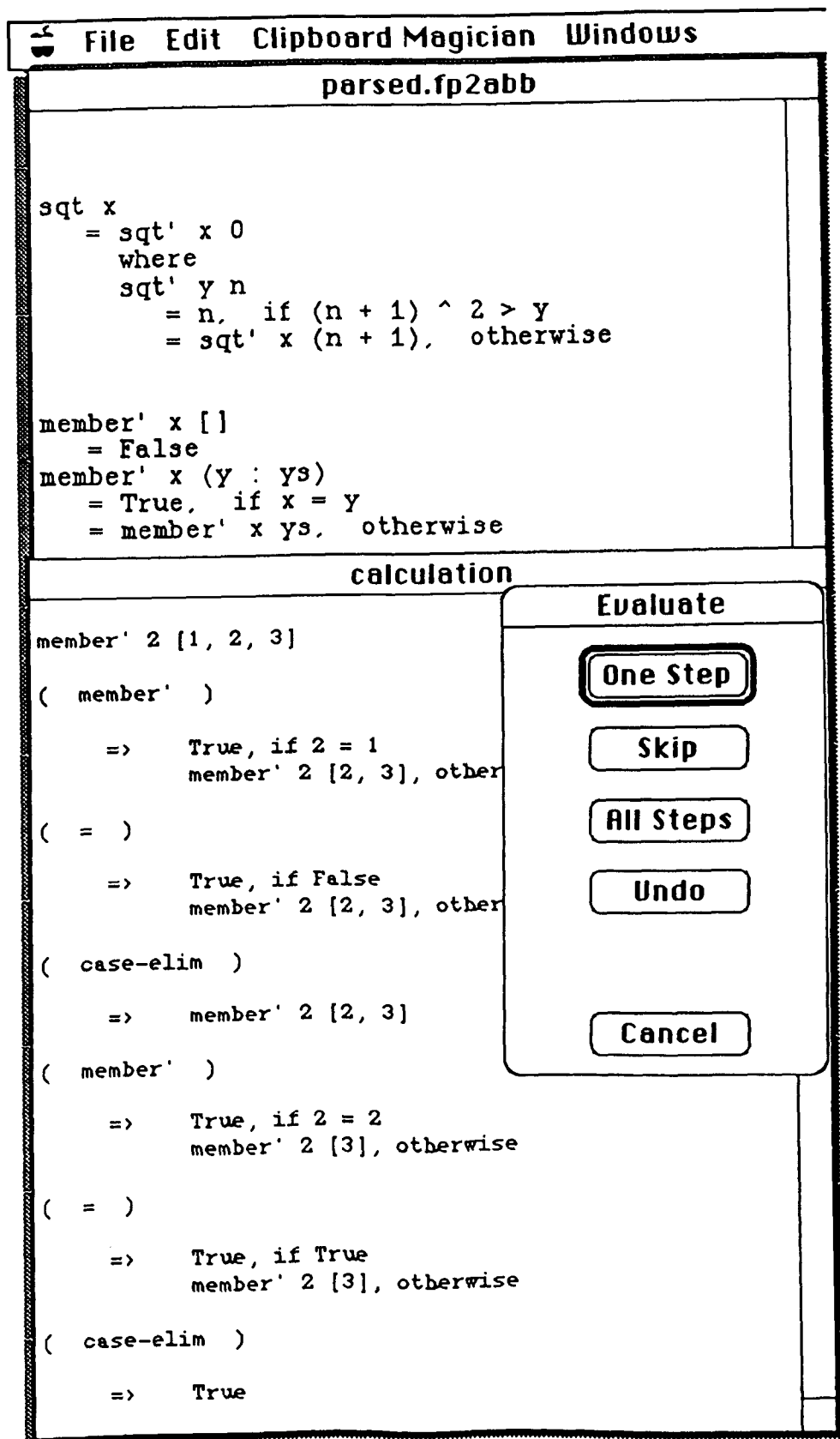


Figure 2.6: The program MiraCalc

Visualisation tools: Venn

Venn (Chariot Software Group, California) supports the teaching of syllogisms by constructing Venn diagrams (Figure 2.7).

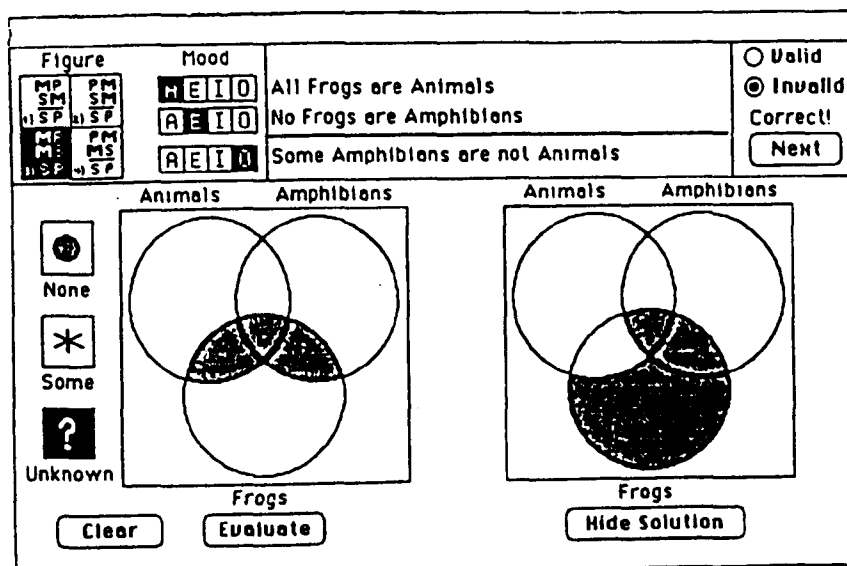


Figure 2.7: The program Venn

Like MiraCalc, Venn tries to support learning by using examples. In this case, however, the examples are sentences in natural language which are selected from a database integral to the program. These are investigated by constructing Venn diagrams, which are used to determine the validity of particular inferences.

This software's design has much in common with that of the previous two packages. It automates book-work in order to promote exploration, practice and experience. Like MiraCalc, Venn attempts to provide a context for this process by using examples. What distinguishes it this from other tools for exploring applications of logic is that it uses a graphical representation to allow students to visualise the structure of the proof.

Learning Environments: Tarski's World

Learning environments form the fourth category of logic-learning software. This area is dominated by two related tools: Tarski's World (Figure 2.8; Barwise & Etchemendy, 1992) and Hyperproof (Barwise & Etchemendy, 1994).

The Tarski's World interface has three main elements. The first of these is the World window. This displays an 8x8 grid, on which pieces can be placed. These pieces may be one of three shapes (tetrahedrons, cubes, or dodecahedrons), and may be either small, medium or large. In addition, each piece may have one or more names. These worlds can either be created from scratch, or loaded in from a directory of saved examples.

The second window is the sentence editor. Here, sentences are constructed in an extended version of first-order logic. In addition to the standard logical operators, a variety of predicates are defined. These include one-place predicates such as assertions about shape and size (e.g. Large (a), Cube (b)), and two- or three- place predicates which describe spatial relationships (e.g. LeftOf (a,b) and Between (a,b,c)). Checking the truth of these becomes trivial when presented with a completed diagram. Entries in the sentence window can be evaluated to see if they are well formed, if they are sentences, and whether or not they are true (assessed by comparison with

the current World). As with the World window, sentences can either be loaded in or created from scratch.

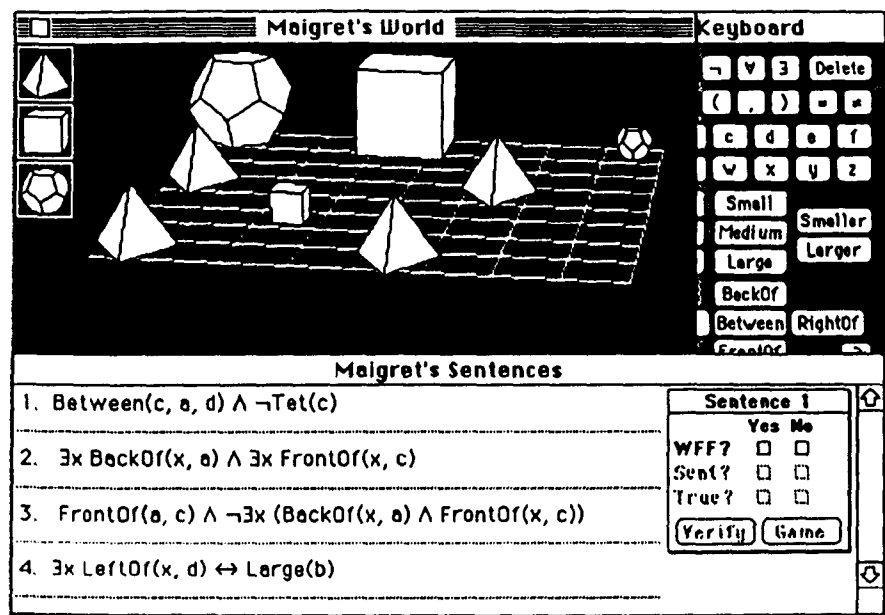


Figure 2.8: a screenshot from the program Tarski's World

The third interface element is the Keyboard window. This palette of buttons allows sentences to be built up using mouse-clicks; this avoids having to type entire words or needing to enter logical notation using a standard keyboard.

A typical exercise in Tarski's World would be to find a way of naming the objects shown in the World window which satisfies the set of sentences displayed in the Sentence window. Disputed results can be worked through using the Henkin-Hintikka game, which evaluates the truth of sentences interactively; this allows students to challenge conclusions by attempting to logically justify their alternatives.

One criticism of Tarski's World is that it is limited in its capacity to express generalities. Hyperproof (Figure 2.9) developed the blocks-world metaphor in order to introduce ambiguity (Barwise & Etchemendy, 1990). It did this with graphical conventions capable of hiding the shape, size and position of a piece (a bag, a cylinder, and placing the piece to one side of the board, respectively). It also supports open-ended problems such as, "what is the most that can be said about the number of cubes left of d ?".

Tarski's World has been shown to be effective at helping students overcome the conceptual and motivational problems associated with first-order logic (e.g. Fung & O'Shea, 1994; van der Pal, 1995). Using interactive graphics in this way has also been shown to support transfer to Wason selection tasks (van der Pal, 1996). Features which were identified as useful for students included being able to check the correctness of proofs, support for book-keeping activities, and combining the formal notation with a simple graphical domain.

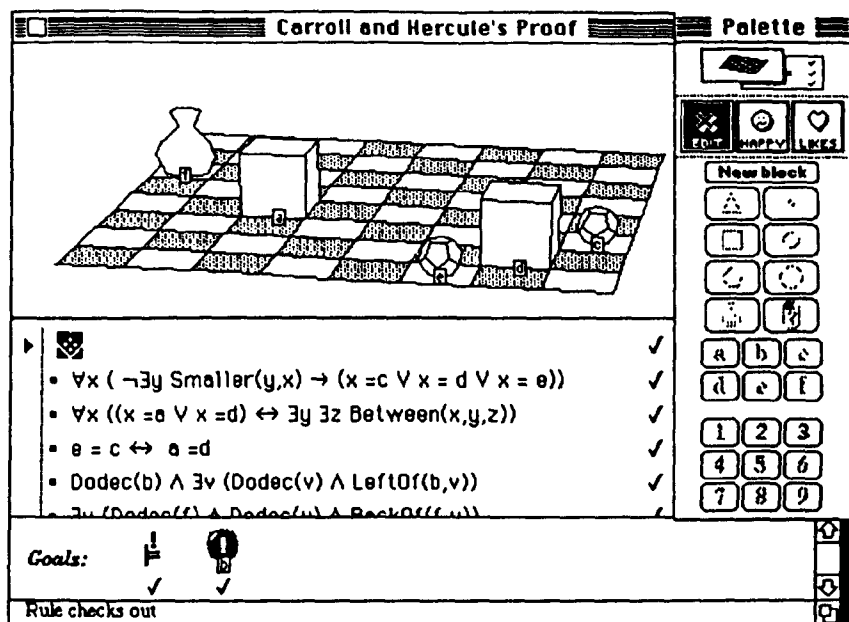


Figure 2.9: A screenshot from the program Hyperproof

It could be argued that tools like MiraCalc also provide a learning environment (in the case of MiraCalc, functional programming), and so should not be distinguished from tools in this category. However, what sets Tarski's World apart is that the blocks-world learning environment is a directly manipulatable microworld which is self-contained and rigorously well-defined. The reason that this is so important is that the reasonableness of solutions to examples grounded in real-world situations (e.g. functional programming tasks for computer science students using MiraCalc) is assessed by comparing the answer to existing beliefs (Campbell et al, 1995). Such beliefs can be context dependent, inconsistent, and difficult to alter (Draper et al, 1992).

Summary

Four key types of logic teaching software (proof construction tools, tools for exploring applications of logic, visualisation tools, and learning environments) have been identified and discussed. In spite of the differences between the tools, they share the common aim of promoting exploration through the reduction of 'book-keeping' activities such as entering logical symbols or building proofs. Another theme, shared by all but the proof construction tools, was the provision of examples. This feature makes software such as Tarski's World more motivating; it also allows programs such as MiraCalc to demonstrate that formal methods, which are often perceived as 'irrelevant', have meaningful applications (Fung & O'Shea, 1992).

2.4 Graphical representations

Many claims have been made for the role of graphical representations in learning and problem solving. Whilst some of these are targeted directly at their place in proofs (e.g. Tennant, 1986), the majority are more general.

This section will summarise the proposed advantages and disadvantages of using graphical representations, mention related cognitive issues, discuss classification systems that allow the

comparison of representations, and close by considering some of the issues involved in combining representations.

Advantages and disadvantages

Many advantages have been proposed for using graphical representations to support learning and problem solving, sometimes with little more than anecdotal support. Such claims tend to make sweeping assumptions, for example that graphics will be better than sentential representations, or that animated diagrams are more effective than static ones, without making reference to particular tasks, domains, or users (Scaife & Rogers, 1995, 1996; Green & Blackwell, 1996).

However, graphical representations are clearly well-suited to use in certain situations. They have been shown to be effective at providing content overviews, at least for expert users (Petre & Green, 1990). They also support clarifying features such as secondary notation (Petre, 1993), and are well suited to preserving or representing topology, geometry, relationships, and chronological or hierarchical structures (Sloman, 1971; Larkin & Simon, 1987; Barwise & Etchemendy, 1990; Chadrsekaran et al, 1993).

Almost any representation can be argued to reduce the load working memory, by virtue of the fact that important information can be recorded externally rather than internally. However, using graphical representations for this purpose has several distinct advantages. For example, if the topological and relational constraints of diagrams are used effectively, searches for information can be simplified (Larkin & Simon, 1987; Cox & Brna, 1995).

More specifically, diagrams seem to offer support for problem solving. Although it has been argued on more than one occasion that diagrams have no role in proofs (e.g. Tennant, 1986), the credibility of such a position has been challenged theoretically (e.g. Sloman, 1971; Barwise & Etchemendy, 1990) and empirically by studies which have demonstrated that graphical representations:

- curtail abstraction, resulting in a weakly expressive language which makes inferences tractable (Cox, Stenning & Oberlander, 1994),
- introduce situativity in domain (van der Pal, 1995),
- replace sequences of complicated mathematical inferences with simpler perceptual ones by using topological relational constraints (Cheng, 1996c),
- provide motivation, and support the process of interpreting what proofs mean (Fung & O'Shea, 1994).

However, using graphical representations also brings disadvantages. Relationships encoded using the topological properties of diagrams can easily be misinterpreted (Hayes, 1993), as can perceptual clues such as symmetry or grouping (Petre, 1993; Petre & Green, 1993). These considerations can easily be extended to form a more general argument: that any perceptual clue, if misinterpreted, will act as a mis-cue, and so interfere with representation use or learning.

There are also many things that graphical representations seem ill-suited to doing, such as supporting abstraction, giving descriptions, or providing lists (Petre & Green, 1990). They can also be slower to work with (Petre & Green, 1993). Perhaps most importantly, there are considerable problems for novice users of graphical representations. Such users often lack appropriate inspection strategies, which can, for example, leave them insecure as to whether or not they have read the diagram thoroughly (Petre & Green, 1993). There is clear evidence that students need to learn how to read and use graphical representations (e.g. Larkin & Simon, 1987; Petre, 1993; Petre & Green, 1993).

In addition to prior experience, another factor which influences whether students use graphical representations effectively is cognitive style (Oberlander et al, 1994). For example, a study with the program Hyperproof (see section 2.3) identified two contrasting styles of problem solving. The first of these involves shorter but more abstract proofs, and is referred to as “DetLo” (indicating a low-scorer on determinate problems); the second involves longer proofs characterised by the exhaustive consideration of cases, and is referred to as “DetHi” (indicating a high-scorer on determinate problems). “Determinate” problems are those best solved by constructing an external representation such as a table or diagram. These styles have also been referred to as “model-hi” and “model-lo” (Cox et al, 1995).

DetHi students seem better able to make use of graphical abstractions during reasoning, and have more flexible proof styles (Oberlander et al, 1994). Interestingly, teaching DetHi students syntactically decreased their performance. Using Hyperproof was more effective for these students, although it did little to help their DetLo counterparts (Cox et al, 1994). A study investigating the use of work-scratchings suggested that DetLo students make less use of external representations to support reasoning; it also seemed to clarify the difference between the two styles as being less about ‘visualisers’ and ‘verbalisers’ than about ability to achieve abstraction with a representation, through the construction of an appropriate semantics (Cox et al, 1995).

Unsurprisingly, then, there are both advantages and disadvantages to using graphical representations. In addition, individual differences between users will need to be taken into account when deciding whether or not a representation is suited to a task.

Systems of classifying graphical representations

Selecting a representation which is well-suited to a task is not a trivial matter, since there is a variety of opinions about how graphical representations should be compared and categorised. However, there is broad agreement that, at the most general level, representations can be compared in terms of informational and computational equivalency (Larkin & Simon, 1987). If two representations are informationally equivalent (i.e. they contain the same information), then whichever is computationally easier to use should be adopted.

This broad approach seems sensible, but fails to provide the means of categorising and comparing informationally equivalent representations. Several methods have been suggested, however. This section outlines six of them: representational type, functional classifications, fidelity, specificity, dimensions and factors influencing design, and isomorphism.

Representational type

One approach, reported on by Cox & Brna (1995), involves classifying different representations by type. This process resulted in 16 categories: natural language, formal language, tables, lists, graphs, plans, set diagrams, graphical tables, time charts, networks, structure diagrams, process diagrams, maps, cartograms, icons and pictures.

Whilst this classification is fairly comprehensive, and useful at a general level, it has drawbacks: the groups cannot be guaranteed to be either exclusive or exhaustive; the allocation of representations to categories is subjective, so that only expert representation users are likely to allocate examples successfully; no provision for comparison within categories is provided; and so on.

Functional classification

Cheng's functional classification provides a more sophisticated means of categorising representations. This scheme attempts to link the properties of representations to cognitive processes. Whilst this might superficially appear to group representations by type, the process by which this takes place is better defined and more detailed, and focuses on a more subtle consideration: representation use. Twelve functions have been identified (Table 2.1), some of which are closely related. As with representational type, however, no guarantee is given that these functions will be exhaustive.

Table 2.1: A functional scheme for describing representations

Function	Description	Example
F1	Showing spatial structure and organisation	Cut-away drawings
F2	Capturing physical relations	Stylized systems of connections
F3	Showing physical assembly	Instruction manuals for lego
F4	Defining and distinguishing variables, terms and components	Component symbols in electrical circuit diagrams
F5	Displaying variables	Pie charts
F6	Depicting states	Schematic diagrams of default switch states
F7	Depicting state spaces	The periodic table
F8	Encoding temporal sequences and processes	Placing diagrams in an ordered sequence
F9	Abstracting process flow and control	Data flow diagrams
F10	Capturing laws	Force diagrams in physics
F11	Doing computations	Nomograms
F12	Computational sequencing	Graphs being used to explain integration

Fidelity

Other approaches to classifying representations do so by introducing one or more 'dimensions'. One simple approach is to compare representations by 'fidelity' (Friedman, 1993). This involves ordering representations according to how 'realistic' or 'abstract' they are. Figure 2.10 summarises this scheme.

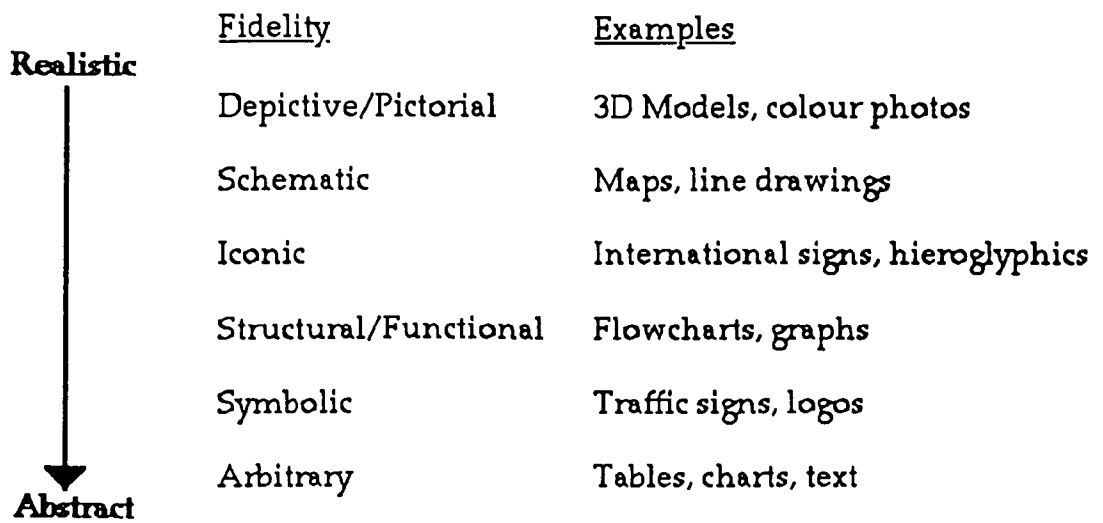


Figure 2.10: Comparison by fidelity

Whilst this scheme allows representations to be compared, it fails to identify what the best application of a representation might be, whom for, or why. This makes it a primarily descriptive theory.

Specificity theory

A related but more successful approach uses 'specificity', which mainly considers representations used in the domain of problem solving (Stenning & Oberlander, 1995). This classifies systems as being capable of expressing minimal, limited, or unlimited abstraction (MARS, LARS, and UARS, where 'RS' denotes 'representational system'). These are explained as follows (Table 2.2):

Table 2.2: Degrees of specificity

MARS	Each individual representation describes precisely one model
LARS	Representations abstract over a number of models, with each sub-diagram describing precisely one model. This can be achieved through combining multiple diagrams or introducing conventions for abstraction.
UARS	These express dependencies either inside a representation, e.g. by equations, or outside the representation, e.g. by interpreting key assertions.

It is argued that this account allows a computational description of the processing differences between systems, with MARSs and LARSs supporting more tractable inferences than UARSs. Representations can then be chosen which are suitable for particular types of problem solving. 'Suitability' is determined by comparing the expressivity of the representation to the expressivity required by the problem.

To illustrate this, MARSs will be suitable for fully determinate problems, and LARSs (or possibly UARSs) for abstract ones. This classification is particularly useful when choosing representations for problems sufficiently determinate to motivate the use of graphics, but which involve enough abstraction to make a simple disjunction of MARSs impractical. For example, Hyperproof places pieces off of the edge of the board when their location is unknown, using this as a graphical convention to denote abstraction. The alternative to this LARS would be to have a disjunction of up to 64 MARSs, each denoting one possible location, for each piece whose location was unknown.

Dimensions and considerations for the design of external representations

A set of considerations for representation design has been proposed by Scaife & Rogers (1996). These include two elements which can be thought of as dimensions, in addition to more general design issues. These dimensions are:

- **Explicitness and visibility:** The ability to facilitate parsing and making inferences by directing attention to useful components.
- **Cognitive tracing and interactivity:** This dimension describes what appears to be a trade-off between different forms of representations. Diagrams, for example, provide good support for cognitive tracing (e.g. through animation); microworlds and simulations lose this, but support interactivity.

The more general design issues, which can also be used to distinguish between representations, are:

- **Ease of production:** If representations are easy to make, more practice can be gained with constructing them, which should support subsequent interpretation.
- **Combining external representations:** Many representations involve other graphical and textual elements, expecting them to be used to support each other. These need to be integrated in order to be effective. However, representations using different modalities, and even representations separated spatially, can hinder the process of integration.
- **Distributed graphical representations:** How well the representation supports collaborative construction and interpretation.

By focusing on practical issues of usability as a means of comparing representations, these considerations capture several important qualities neglected by the previous systems. By the same token, however, this focus draws attention away from qualities such as how well representations support abstraction, which seem to be important in determining which systems are most suitable for students learning logic.

Isomorphism

A final dimension, which occurs in a variety of forms throughout the literature, is how accurately a representational system describes what it is modelling. In its weakest form, this dimension describes 'analogical' representations, which preserve relationships between elements of the model, so that "the structure of the representation gives information about the structure of what is represented" (Sloman, 1971). Such representations allow inferences to be made for the representation, and then translated into conclusions about the model. Similar to this is the notion of 'good' diagrams. 'Good' representations are defined as being homomorphic to the model (Barwise & Etchemendy, 1990). This means that a well-defined correlation exists between elements in the representation and those in the model. 'Vivid' representations refine this idea still further, requiring an isomorphism from content to representation (Levesque, 1986). As a result, once the translation process is known, the representation can be derived from the model, and vice versa. All of these definitions make use of structural constraints and the explicit

representation of abstract relationships to direct attention to salient features and support cognitive processes.

These ideas have been explored further with law encoding diagrams. Here, each diagram correctly encodes the underlying relations of a law, or system of laws, by means of geometric, topological and spatial constraints, so that each instantiation of a single diagram represents an instance of the phenomenon or one case of the laws (Cheng, 1996d). These constraints limit interpretations, making discoveries more tractable than if algebraic relationships had been used (Cheng, 1996c). This is argued to make learning about laws and phenomena easier (Cheng, 1996d). The effectiveness of these representations has been demonstrated for students learning about two-body collisions, where the diagrams made it easier to explore the space of possibilities, and hence gain a better qualitative understanding of the domain (Cheng, 1996b).

It is possible to have several law encoding diagrams for the same system of laws (Cheng, 1996c). As with the initial problem of distinguishing between informationally equivalent representations, the effectiveness of different law encoding representations depends on the computational complexity of the representations (Cheng, 1996d).

The relationship between classification systems

Whilst these six systems of dimensions, classification and design considerations are distinct, there remain several areas of overlap. Some of the classification systems can be considered to be complementary; for example, one argument for the effectiveness of law encoding diagrams is that they combine many of the functional roles of diagrams in one representation (Cheng, 1996a). Although each system has limitations, between them they provide a variety of techniques which can be employed to distinguish between representations in terms of their efficiency.

Summary

In this section, the advantages and disadvantages associated with using graphical representations have been reviewed. These considerations suggest that representations need to be carefully selected if they are to be effective for learners. This selection process must consider users' prior experience, cognitive style, and the task requirements.

Six systems for comparing representations have also been considered. Whilst each of these has limitations, they provide a useful variety of methods for comparing representations.

2.5 Complementary ways of using graphical representations

Even a cursory inspection of representations shows that no one 'type', as identified by any of the systems described in section 2.4, is universally the most effective at supporting learning. This is not surprising: different tasks will require different features from a representation, and any given representation can be used in a multitude of ways. Instead, effective representations seem to 'specialise' in sub-sets of the qualities described above in order to meet specific task requirements.

To establish which specialisation best support students learning formal reasoning, it is necessary to determine which tasks the representation is required to support. It has been proposed that visual information can play one of three roles in reasoning (Barwise & Etchemendy, 1990):

- i) It can form part of the given information from which we reason
- ii) It can be integral to the reasoning process
- iii) It can play a role in the conclusion of a piece of reasoning

A very similar classification motivated the development of law encoding diagrams (Cheng, 1996d):

- i) Representations can be used to simulate the surface-level properties and behaviour of systems
- ii) Representations such as graphs can be used to make complex information easier to understand
- iii) Instructional systems can use graphical representations to explicitly reveal or organise the way information is structured within a system.

Both of these classifications can be simplified by combining two categories (i and iii from the first system; ii and iii from the second). In both cases, this results in essentially the same remaining pair of uses for visual information in this context: as content for, or to support, the process of reasoning. (These will be referred to as 'content' and 'process' representations respectively.)

It is proposed that these roles are complementary. There would be no informational redundancy between representations in the two different categories. One would simply identify whether or not atomic propositions are correct, whilst the other would model the way in which these propositions could be combined and manipulated.

Furthermore, these two types can be categorised differently on each of the six systems of classification and comparison described in the previous section. For example, by type, the content representation is likely to be something such as natural language, a map, or a picture; the process representation is far more likely to be part of one of the remaining categories, such as a formal language or a process diagram.

There will, of course, be exceptions to this categorisation. For example, if someone tries to deduce the rules of syllogistic reasoning from Venn diagrams, the content representation will be of the type, "set diagram". Being formal, abstract, and de-contextualised, however, it is easy to argue that this representation will make the process of deducing the laws of syllogistic reasoning difficult. For the purpose of this thesis, however, representations will be classified as content or process representations according to the type of reasoning tasks they are best suited to supporting.

As mentioned above, content and process representations remain distinct on all but one of the systems of comparison described in 2.4. The exception is on the dimension of isomorphism. Even here, there is a difference: content representations will need to be isomorphic to an example,

whilst process representations will need to be isomorphic to a process. However, the law-encoding isomorphism of the content representation is likely to be trivial — such as using perspective drawings to encode a spatial relationship. Table 2.3 summarises how exemplars of content and process representations can be categorised in each of the systems.

Table 2.3: A comparison of representations for content and process representations

System	Content representations	Process representations
Type	Natural language, maps, plans, cartograms, icons, pictures	Formal languages, tables, lists, graphs, set diagrams, graphical tables, time charts, networks, structure diagrams, process diagrams
Function	F1 (spatial structures), F2 (defining variables), F5 (displaying variables), F6 (depicting states)	F2 (physical relations), F3 (physical assembly), F7 (depicting state spaces), F8 (encoding temporal sequences), F9 (process flows), F10 (capturing laws), F11 (doing computations), F12 (computational sequencing)
Fidelity	Realistic	Abstract
Specificity	MARS	LARS or UARS
Design considerations	Interactive	Supports cognitive tracing
Isomorphism	Isomorphic to an example	Law encoding (for the reasoning process)

The intuitive content/process distinction, although simple, is robust. The same distinction between types of representation can be made for each of the six systems of classification. This means that it is neutral with respect to the different approaches to classifying representations, making it a useful common ground for discussions in a field riddled with varied terminology.

Inspection of Table 2.3 also gives support to the idea that representations with different ‘specialisations’ can be classified in terms of which of the two distinct roles they are best suited to fulfilling.

This classification predicts that a concrete, pictorial representation with minimal expressivity that supports interaction will be well suited to providing content to the reasoning process. Similarly, it suggests that representations which encode the rules of inference or manipulation (and hence such features as temporal sequences or process flows), support cognitive tracing, and which are adequately expressive (i.e. a LARS or UARS, and hence abstract to some degree) will be effective for supporting the process of reasoning.

Furthermore, since content and process representations seem to be complementary in both form and function, it seems likely that using them in parallel will prove effective. This integration would involve several of the uses identified for multiple representations (Ainsworth et al., 1996). The two would support different processes, present different information, and each would use the other representation to constrain interpretation. The content representation would provide a context for the process representation, whilst the process representation would direct the user to salient features of the content representation. This can be illustrated by an example consisting of a line linking a series of related states from a game of chess. Following the line and noting the change in states would imply that some situations follow others, constraining an interpretation of the moves which have taken place. Taking the other possible perspective, consideration of

the states themselves would reveal that changes have occurred and the game has developed, implying that the line represents a chronological ordering of some sort.

Using two representations in this way may also address the concern that representations which are good for experienced users aren't necessarily good for students learning about the topic (Good & Brna, 1996). In this case, the content representation may support novices whilst they learn how to use the process representation effectively.

Initial empirical support for this idea can be drawn from the finding that lower ability children learning estimation benefited from using pictorial representations (Cronbach & Snow, 1977; Ainsworth, 1997). Here, the syntactic, symbolic process representation can be seen as being complemented by a concrete, pictorial content representation. This distinction would also account for the finding that versions of Tarski's World with either the sentence window or the blocks-world window disabled were less effective for learners than the complete package (van der Pal, 1995). Similar findings have been observed for microworlds incorporating formal representations in other abstract domains, such as physics (e.g. Alessi, 1988).

Complementary representations for learning Modal logic

Having identified several potential benefits of using complementary representations to support reasoning, it is appropriate to reconsider the two systems of representing Modal logic described in section 2.2 in order to establish whether they already make use of this approach.

Sentential representations code all information syntactically. For this reason, they are precluded from taking advantage of the benefits of complementary representations. By contrast, the diagrammatic style of representation can be argued to use complementary representations. In this case, the process representation consists of the box-and-line diagrams used to provide structure in proofs. The content is represented syntactically, using a variant of the syntactic representation.

Although the diagrammatic condition makes use of complementary representations, the qualities presented in Table 2.3 suggest that a syntactic content element is not ideal. The classification systems reviewed in section 2.4 suggest that a representation which is concrete and pictorial, possesses minimal expressivity, and supports interaction will be more effective for learners. The abstract syntactic notation adopted by the diagrammatic representation fails to meet any of these criteria.

This suggests that a new style of representation for Modal logic could be derived which would be more effective for learners than the existing system. A revision of the diagrammatic representation, with the syntactic content representation replaced by something closer to the 'specialisation' recommended by the classification systems described above, should make the system more effective for learners.

Summary

A simple distinction between representations which provide content and those which support processes has been proposed, based on a consideration of the classification systems described in

section 2.4. This distinction between content and process representations remains constant across all six systems.

Neither the syntactic (Figure 2.3) nor the diagrammatic representations (Figure 2.4) currently used to teach Modal logic make full use of complementary representations. For this reason, a new system can be developed which meets the criteria outline in this section. This system should prove more effective for learners than either of the current representational systems.

2.6 Conclusion

Problems associated with learning Modal logic have been identified by reviewing current teaching texts and exploring the literature on learning first-order logic. This indicated that students who lack prior experience with abstract or formal systems find the process of learning logic to be particularly difficult. Additionally, students often encounter motivational problems. The review also described seven key topics in Modal logic which have the potential to be conceptually difficult for learners, but identified the need for further research in this area.

After discussing the problems associated with logic learning, four ways in which software tools can be used to support this activity were outlined. Examples of programs which characterise each of these approaches were then given. These four approaches shared two common features: the use of examples, and the automation of 'book-keeping' or routine tasks in order to promote learning.

The examples used in two of these tools were essentially graphical. This approach seems best suited to the requirements of examples: that they be clear, self-contained, and illustrative. Consequently, the literature on using graphical representations to support reasoning was reviewed. This led to the specification of two uses for representations in reasoning: either to provide content for reasoning, or to support the reasoning process. The features which characterise these two types of representation use were summarised, and the potential benefits of using them in parallel were discussed.

Reviewing the representations currently used to teach Modal logic shows that neither takes full advantage of the potential benefits of complementary representation use. It is proposed that a third system of representation be developed for Modal logic based on this new approach. This new system will be described in chapter three, together with a study designed to test the prediction that this system will provide better support for learners than the syntactic or diagrammatic representations. This study also offers an opportunity to investigate which topics in Modal logic are conceptually difficult for learners.

Chapter 3

Using complementary representations to support learners of Modal logic

3.1 Introduction

Chapter two reviewed current literature on learning Modal logic and the use of graphical representations to support reasoning. This review led to the creation of a new classification system for representations which distinguishes between those intended to provide content for and those used to support the reasoning process. It was argued that these roles were complementary, and that a system which combined both types would effectively support learners.

Inspection of current representations for Modal logic showed that neither the syntactic nor the diagrammatic styles made full use of complementary representations. It was proposed that a third style of representation be created, based on this new approach.

This chapter presents the new system of representation. It then describes an experiment designed to test the hypothesis that this new system will provide better support for learners than the syntactic or diagrammatic representations.

Section 2.2 identified seven topics in Modal logic with the potential to prove conceptually difficult for learners, and noted that the causes of these difficulties merited further research. This experiment also aims to investigate these problems, and to arrive at a clearer picture of the problems associated with learning Modal logic.

3.2 A new representational system for Modal logic

Identifying the need for a new system

Section 2.5 identified two distinct roles that representations can take to support the reasoning process. These are referred to as content and process representations. Section 2.2 identified two styles of representation currently used to teach Modal logic: syntactic and diagrammatic.

Neither of these systems makes full use of the potential offered by complementary representations. The syntactic style uses only one form of representation. An analysis of the diagrammatic style showed that its box-and-line component meets the criteria for acting as an effective process representation, since it is law encoding, able to support abstraction, and describes the process of proof construction. However, the syntactic content of each box fails to meet the criteria for forming a good content representation. This suggests that whilst the style uses multiple representations, a more effective system could be created by replacing the syntactic element with more concrete representations.

The blocks-world metaphor of Tarski's World has been shown to be an effective support for students learning logic (e.g. Fung & O'Shea, 1993). Additionally, it is concrete, interactive, minimally expressive, and essentially pictorial. These qualities make it ideally suited to use as a content representation (see section 2.5). It is proposed that a new system which combines this with the box-and-lines structure of the diagrammatic representation will provide effective support for learners.

An overview of the new system

Having argued that complementary representations have the potential to provide effective support for learners, and having identified elements ideally suited to acting as content and process elements for Modal logic, it remains only to combine these.

The simplest combination involves substituting illustrations based on the blocks-world of Tarski's World for the formulas in the diagrammatic representation. It should be noted that the metaphor was adapted slightly in order to make the illustrations easier to produce, in line with Scaife and Rogers' recommendations (1996). The revised representation comprises of four types of element: a grid, on which objects are placed, a sphere, which can move around the grid one square at a time, walls, which block the sphere's movement, and letters, with which the sphere's position is compared, using the same propositions as Tarski's World (e.g. LeftOf (a,b) and Between (a,b,c)). An example of a world with all four types of element is shown in Figure 3.1.

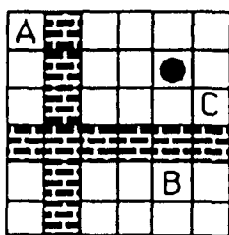


Figure 3.1: A grid world with one circle, two walls, and three letters

Worlds are then connected using the box-and-line element of Hughes & Cresswell's representation (1968). This provides the law-encoding 'process' element of the representation.

As a convention to support abstraction, the existence of possible worlds which aren't explicitly represented is denoted by a box containing the word, "Etc". Figure 3.2 illustrates these conventions for the example, "the circle can move one square".

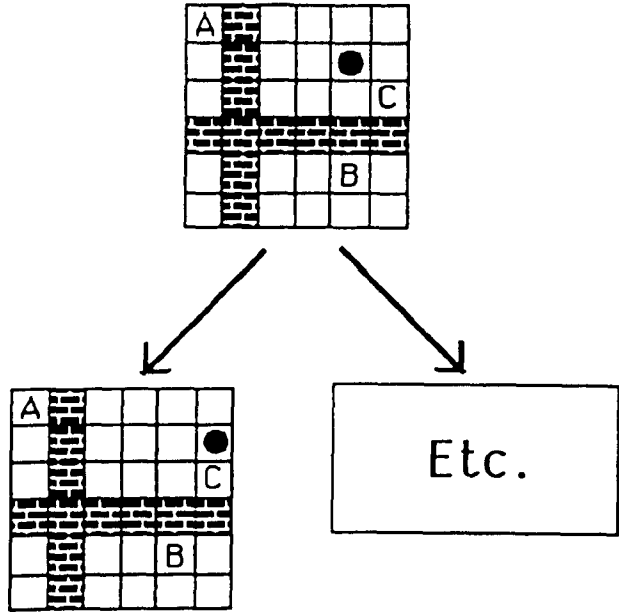


Figure 3.2: An example of a relationship between possible worlds

To show how this compares with the syntactic and diagrammatic representations of Modal logic, an example of three worlds in a transitive relationship is shown for all three representations. Figures 3.3 and 3.4 illustrate this for the syntactic and diagrammatic styles of representation.

$$((W \ R \ W') \ \& \ (W' \ R \ W'')) \rightarrow (W \ R \ W'')$$

Figure 3.3: a transitive relationship, expressed syntactically.

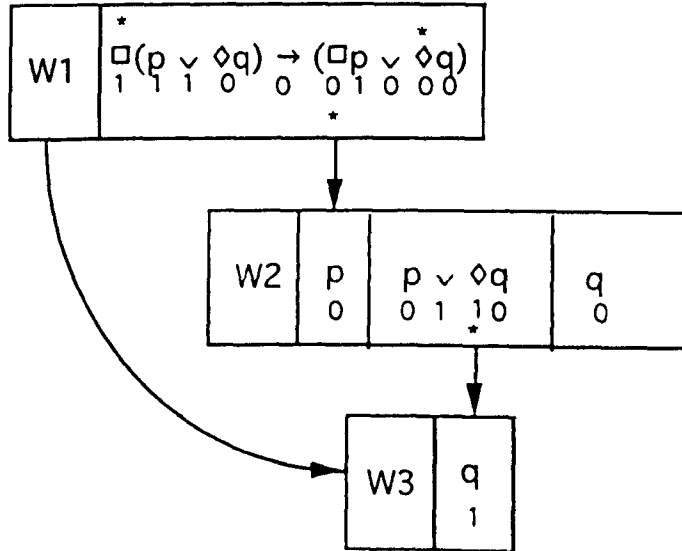


Figure 3.4: a transitive relationship, expressed diagrammatically.

In the new representational style, a transitive relationship would resemble Figure 3.5.

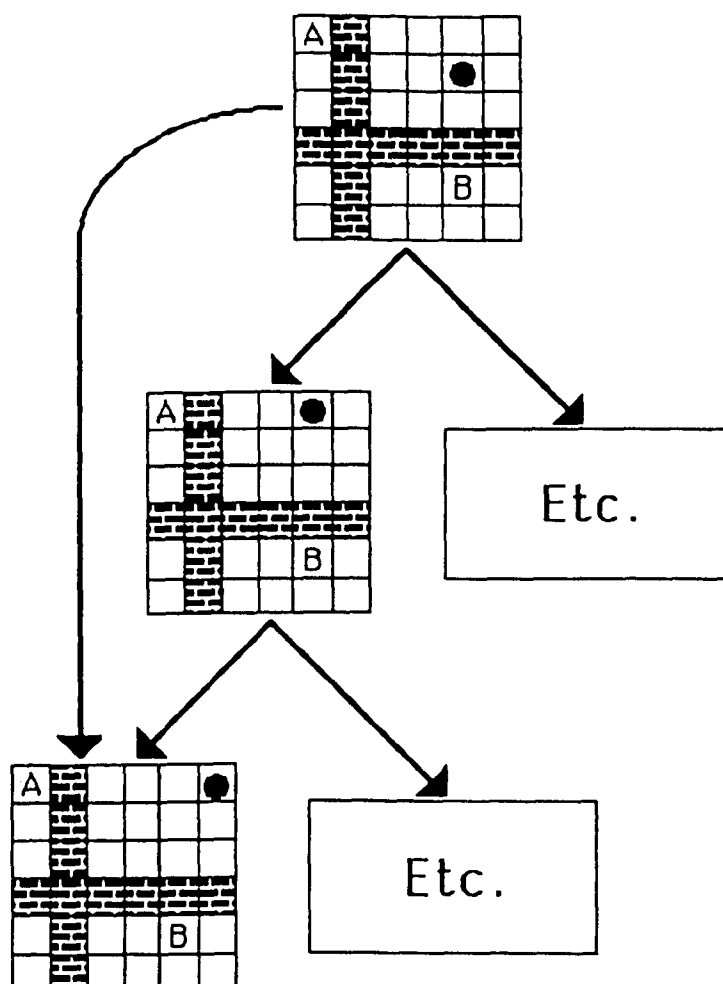


Figure 3.5: a transitive relationship, combining content and process representations

The new system of representation uses arrows to show transitivity in the same way that the diagrammatic style does (Figure 3.4), using the arrow on the left-hand side to show the additional link between the first and third worlds. In this case, the situation represented would be something like, “allow the ball to roll up to two times”.

Since this representation is derived in part from the blocks-world metaphor used by Tarski’s World, and due to the intention to adopt this as the basis of a software tool for Modal logic, it will be referred to throughout this thesis as the “learning environment” style of Modal logic.

3.3 A comparative study of the three representations

In the previous section, a new system of representation for Modal logic was derived. This was motivated by the argument that a system incorporating complementary representations should provide better support for learners than the syntactic or diagrammatic representations which are currently used.

A study was designed to test this argument. By comparing the conceptual improvement of subjects in three groups, each using one of the representational styles, the study also sought to establish how the representations differed in terms of their effectiveness for learners. This also provided an opportunity to investigate which of the seven concepts identified in section 2.2 cause difficulties for learners of Modal logic, allowing the study to address the two areas identified as needing further research in section 2.6.

Methodology

Two complementary approaches were combined in this study. An experimental methodology was adopted to investigate the comparative performance of subjects on quantifiable measures of performance. The improvement on measures tested at both pre- and post-test is reported on in the first results section; the second describes results relating to measures taken at post-test only, such as measures of confidence.

Qualitative data was gathered and analysed in order to gain further insight into the problems associated with learning Modal logic, and into the differences between instructional conditions. A grounded theory approach to qualitative data analysis (Glaser & Strauss, 1967) was adopted for responses to open questions and for interview data. This led to the development of categories argued to be of central importance to understanding the problems associated with learning Modal logic.

Hypotheses

It was assumed that all seven of the concepts described in section 2.2 would prove difficult to learners. Given that section 2.2 also identified motivation as a problem for students learning logic, self-assessed measures of enjoyment and confidence in performance were also investigated.

Two of the concepts from section 2.2, reference and belief, are too extensive to be investigated in fully in this study. Rather than neglect these completely, the instructional material addresses each briefly. Subjects were assessed on their awareness of issues which related to these concepts, rather than on the complex mechanics of their formal applications.

It was hypothesised that the syntactic instructional condition would be the least effective for learners on all of these measures, and the learning environment condition the best of the three. Effectiveness would be measured by comparing pre- and post-test measures of understanding. A similar pattern was anticipated for the motivational benefits of the three instructional conditions.

This pattern was proposed for two reasons. Firstly, law encoding diagrams have been shown to be effective for students learning abstract topics (Cheng, 1996b). Since the diagrammatic and learning environment instructional conditions make use of these, and the syntactic condition does not, this suggests students will find the former two instructional conditions more effective than the latter. Secondly, the learning environment condition was designed to take advantage of complementary representations, whilst the diagrammatic condition was not. This suggests that the learning environment representation will offer better support than the diagrammatic representation.

The length of time taken to complete the intervention material was also noted. It was anticipated that the syntactic condition would be the quickest, and the learning environment material the slowest, for two reasons. Firstly, the learning environment condition required students to learn about the block world that was being used for examples in addition to learning about Modal logic. Secondly, the learning environment material was expected to provoke

reflection. This also suggests that the length of time taken to work through the material would be positively correlated with improvement.

Subjects

A total of 47 subjects were drawn from three sources: staff from the Open University, Computer Science students from Hertfordshire University, and Computer Science students from Aberystwyth University. All subjects were volunteers, and those from Hertfordshire and Aberystwyth were paid for participation.

Materials

The intervention material used in this study drew on chapter 8 of Reeves & Clarke (1990), together with chapters 2 to 4 of Hughes and Cresswell (1968). This sufficed to introduce all seven concepts. Three sets of material covering the same concepts and topics were created, each using a different style of representation. These were pilot tested with three participants, each of whom worked through one set of material and then compared it with the other two. The results of this pilot were used to refine the teaching material and ensure consistency between the three instructional conditions.

Since the first section covered intuitive concepts which were not easy to express formally or diagrammatically, this part of the material remained unchanged for all three instructional conditions.

Pre- and post-tests were designed which would assess subjects' understanding of the seven key topics from Modal logic described in section 2.2. Additionally, a near-transfer of knowledge question was added which required subjects to apply what they knew to a new system of Modal logic.

These questions were based on questions from two sources: chapter 8 of Reeves & Clarke (1990), and material from the 1995 Modal logic course at the Department of Mathematics and Computer Science, University of Leicester. The questions used to assess performance in these topics are identified in Table 3.1.

Table 3.1: The pre- and post-test questions used to assess the concepts under investigation

Concept or topic	Questions used to assess performance
Use of Modal operators	Question 2
Possible worlds	Questions 1 (a) and (b)
Reference	Question 1 (d)
Belief and Modal operators	Question 1 (c)
Modal systems	Question 3
Modal proof construction	Question 3
Near transfer	Question 4

All three instructional conditions used the same pre- and post-test questions, expressed formally in order to ensure consistency between the groups. Students were permitted to answer the questions using whichever style of representation they preferred.

Additionally, the pre-test gathered categorical data on the background of each respondent, covering prior experience with logic and gender. The post-tests asked the respondent to assess

how well they had performed, and to say which topics had been problematic. Subjects were asked to indicate on five point scales how much they had enjoyed the material and how confident they were that they had improved, relative to the pre-test. These data were used to assess the motivational effects of the three instructional conditions. Open questions targeted three specific areas (the problem of reference, proof construction and the transfer question) anticipated as being particularly problematic. Finally, this section invited comments on the teaching material, and (after providing them with an example page from both other instructional conditions) asked them for a preference towards one of the three representational styles.

At all times, the subjects were able to refer to a glossary of symbols and a summary of the rules of first-order logic. These were included to reduce the bias that a lack of familiarity with first-order logic or the notational system would cause. All materials used in this study can be found in Appendix 1.

Design

The study set out to investigate a series of measures of performance and of motivation, taking into account factors of instructional condition, prior experience with logic and gender.

Instructional condition was proposed to be a factor based on the theory outlined in section 2.5 which led to the derivation of a new style of representation in section 3.2. This suggested that the learning environment condition would prove most effective for learners, and the syntactic condition the least.

Prior experience with logic was identified as a factor influencing logic learning in section 2.2.

Gender differences have been found in many scientific, mathematical and logical areas (e.g. Lovegrove & Hall, 1991). For this reason, gender was also selected as a possible factor influencing learning outcomes.

This resulted in a mixed experimental design, involving a series of within group repeated measures of performance and between groups factors of instructional condition, experience and gender.

The dependent variables were measures of performance on the concepts identified in section 2.2. Subjects' understanding of possible worlds, of the interaction between Modal logic and beliefs, their ability to construct proofs, and their success on the near transfer question produced category data. The remaining measures scored subjects on the number of Modal systems explained or used correctly.

Procedure

47 subjects were randomly allocated to the three instructional conditions. These conditions differed only in the intervention material they used; the instructions, glossary sheet, pre- and post-tests were identical in all three. Materials were self-administered, and subjects were required to note their start and finish times.

Each subject completed a pre-test, then worked through the instructional material. When they were satisfied that they had learnt as much as they could from this, they attempted the post-test.

Seven subjects from each instructional condition worked through the material under observation. These received the second part of the post-test (as described in the materials section) in the form of a semi-structured interview which gave them the opportunity to expand on their responses. The data gathered from these subjects was used to complement and expand upon the written responses of their unobserved counterparts.

Answers for the first part of the post-test were compared with the equivalent pre-test questions to measure learning gains. The open questions in the second part of the post-test provided qualitative data. Of the 15 face-to-face subjects, one was asked to participate in an in-depth interview following the experiment. An analysis of this is detailed in the case study section.

3.4 Results: Pre- to post-test measures

This section contains results from measures of performance taken at pre- and post-test. These are preceded by a summary of the distribution of subjects.

The Distribution of Subjects

As noted, subjects were randomly assigned to each of the three instructional conditions. Table 2 summarises the distribution of respondents with regard to the independent variables identified in the design section, and indicates the distribution of observed and unobserved subjects. Full background data, together with the data for the various measures of performance, can be found in Appendix 2.

Table 3.2: group comparison

	Syntactic	Diagrammatic	Learning Environment
Number of Males	9	7	7
No. of Respondents with Moderate FOL experience	7	6	7
Number familiar with Non-Standard logic	3	3	3
Postal Responses	9	8	9
Total in condition	16	15	16

In order to assess the effects of being paid on performance and enjoyment, measures of ability on operator reduction, number of Modal systems learnt, the time taken to complete the intervention material and subjects' rating of how much they enjoyed the study were investigated. Table 3.3 summarises pre- and post-test means and variances for these questions. Scores on the operator reduction question ranged between 0 and 12, the number of systems understood ranged from 0 to 3, the time taken was measured to the nearest five minutes, and enjoyment was scored using a five-point scale.

Table 3.3: means and variances for measures of performance

	Operator Reduction		No. of systems		Time taken	Enjoyment
	Pre-test	Post-test	Pre-test	Post-test		
Paid	3.47 (19.12)	5.67 (20.1)	0.33 (0.81)	1.67 (1.81)	4.73 (8.5)	4 (0.29)
Not paid	3.56 (15.16)	3.81 (16.16)	0 (0)	1.41 (2.12)	4.94 (5.74)	3.44 (2.19)

These results were analysed using an [2x2] ANOVA, with a within groups factor of time and a between groups factor of whether or not the subject had been paid for participation. Results revealed an effect of time on the number of systems learnt ($F(45,1)=35.54$, $p<0.001$) and a trend for an effect of time on ability with operator reduction problems ($F(45,1)=3.98$, $p<0.052$), but no effects of having been paid.

Use of Modal operators

A question was designed which assesses subjects' ability to use their knowledge of Modal systems to solve operator reduction problems. These involve canceling down strings of Modal operators in order to deduce the conditions under which two claims are equivalent.

Subjects answered four such questions, each with three answers required (one for each Modal system). This gave a range of possible scores from 0 to 12 at both pre- and post-test. Subjects who did not attempt the question at either pre- or post-test were excluded from this analysis.

Table 3.4 summarises means and variances for subjects on this question.

Table 3.4: Means and variances for scores on operator reduction

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Pre	3.33 (33.3)	2.5 (12.5)	3.17 (2.07)	5 (25)	8 (0)	8.33 (4.33)	8.5 (0.5)	5.33 (1.37)	6.5 (18)	4 (21)	6.75 (26.3)	4 (0)
Post	7 (7)	1.5 (4.5)	7.7 (12.1)	10.5 (4.5)	3 (0)	5 (19)	5 (0)	1.67 (8.33)	7 (12.5)	7.33 (14.3)	8.75 (4.92)	12 (0)

Results were analysed using an [2x3x2x2] ANOVA, with a within groups factor of time and between groups factors of instructional condition, expertise and gender. Results show a trend for improvement over time ($F(35,1)=3.4$, $p<0.074$) and an interaction between instructional condition and time ($F(35,2)=9.99$, $p<0.001$). There were no effects at pre-test, but an ANOVA on the post-test scores confirmed an effect of instructional condition ($F(29,2)=5.545$, $p<0.009$). Tukey tests were used to compare the mean scores of subjects using each of the representations. These revealed that subjects in the learning environment condition performed better than those in the diagrammatic condition ($p<0.05$).

Figure 3.6 shows the average scores on the pre- and post-tests, showing how subjects' scores in the diagrammatic condition deteriorated. Inspection of the data confirms that every subject in this instructional condition who attempted to answer this question had worse post-test scores than they did pre-test.

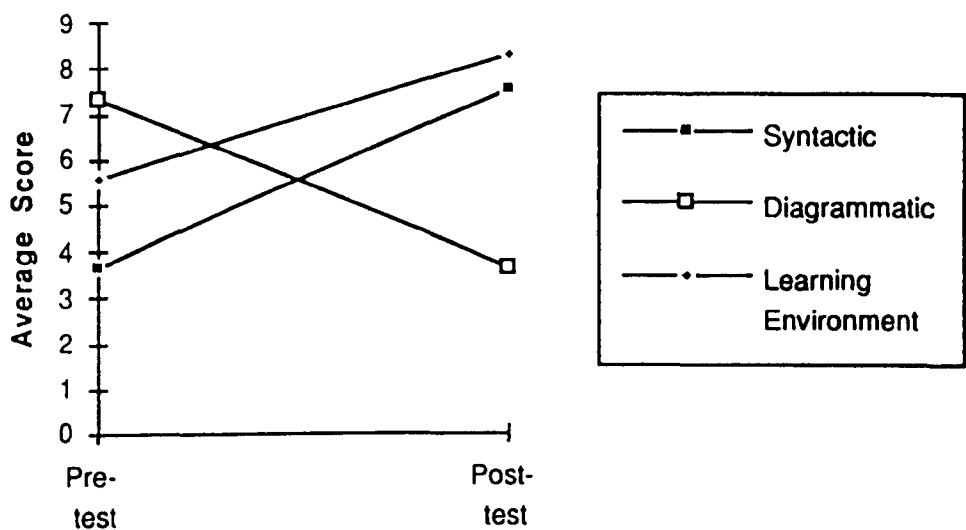


Figure 3.6: Comparison of average scores on the operator reduction question

Attempting operator reduction

Inspection of pre- and post-tests shows that many subjects did not attempt the operator reduction problem. Table 3.5 shows the numbers from each group who attempted the question at either pre- or post-test.

Table 3.5: Numbers of subjects who attempted the operator reduction question

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
No.	3	1	6	2	1	3	2	3	2	4	4	1
Total	3	5	6	2	4	4	3	4	2	7	5	2

A comparison of these data using Chi squared tests showed no effects for instructional condition or gender, but a trend for expertise to improve the likelihood of attempting the question (Chi squared = 3.59, 1 d.f., $p < 0.10$).

Possible worlds

Subjects' understanding of possible worlds was assessed by asking whether a possible world existed in which any of four claims were true. Responses were classified according to whether or not the subject demonstrated that they understood this concept. Table 3.6 summarises the number of subjects who understood the concept at pre- and post-test.

Table 3.6: Number of subjects who understood possible worlds

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Pre	1	3	2	2	1	3	1	1	0	5	2	2
Post	3	4	5	1	3	3	3	3	1	6	3	1
Total	3	5	6	2	4	4	3	4	2	7	5	2

Chi-squared tests revealed no differences between instructional conditions, between novices and experts, or between males and females. McNemar tests were used to compare the pre- and post-test data. These showed a trend for novices to improve on this concept ($p < 0.0654$), whilst experts did not. Males also improved ($p < 0.001$), although females did not. Instructional condition had

no significant effect on improvement. This suggests that performance may have been close to ceiling for subjects in all instructional conditions, with only novices having room to improve.

Reference

Subjects were assessed on their ability to answer questions about the problem of reference. Scores ranged from 0 to 2, with one point being awarded for understanding that either names or symbols can change their meaning. Table 3.7 summarises the means and variances at pre- and post-test.

Table 3.7: Means and variances for scores on the problem of reference

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Pre	1.33 (0.89)	0.8 (0.96)	0.5 (0.25)	1.5 (0.25)	0.25 (0.19)	1 (1.25)	0.67 (0.22)	0.25 (0.19)	0.5 (1)	0.86 (1)	0.4 (0.24)	1.5 (0.25)
Post	1 (0.67)	0.6 (0.64)	1 (0.67)	1.5 (0.25)	0.25 (0.19)	1 (0.19)	1 (0.67)	1.25 (0.69)	0.25 (0)	0.69 (0.29)	1.2 (0.16)	1 (0)

An [2x3x2x2] ANOVA was carried out, with a within group factor of time and between groups factors of instructional condition, expertise and gender. There were no significant main effects, but a trend for an interaction between instructional condition and gender over time ($F(2,34)=2.70$, $p<0.081$).

It appears that the diagrammatic condition accentuated a tendency at pre-test for women to have a better understanding of the concept of reference than men. Figure 3.7 summarises responses to this question.

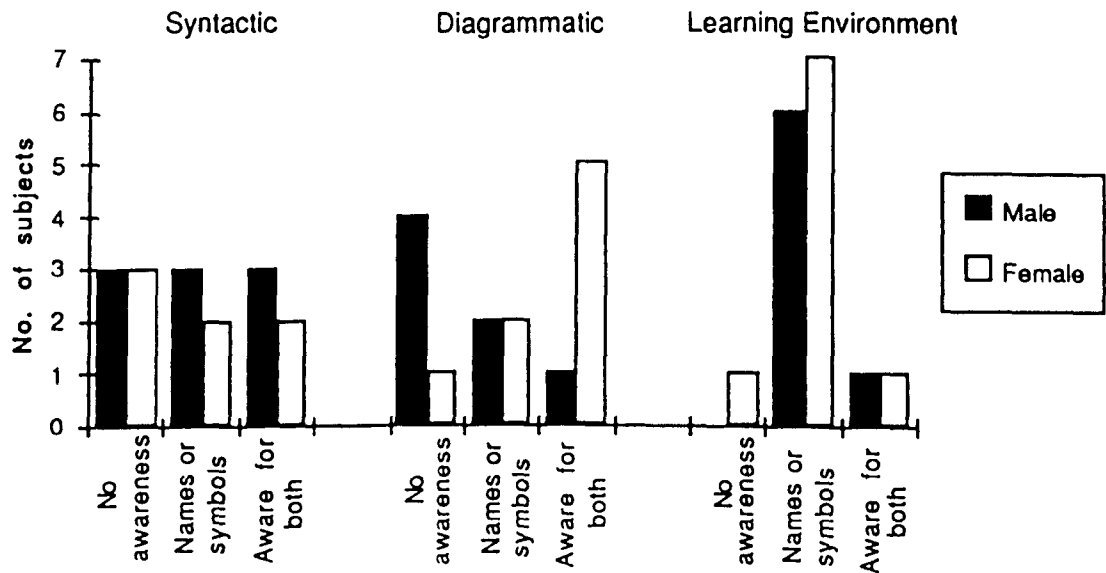


Figure 3.7: A comparison of the responses to the post-test reference question.

Belief and Modal operators

Subjects’ understanding of the ways in which Modal logic interrelates with beliefs was investigated by a question which asked whether it was possible for a hypothetical agent to believe certain claims. Responses were classified according to whether or not the subject demonstrated an understanding of this concept. Table 3.8 summarises the number of subjects who understood the concept at pre- and post-test.

Table 3.8: Number of subjects who understood how Modal logic and belief interact

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Pre	1	4	3	1	1	2	2	3	1	3	3	2
Post	3	4	6	2	3	3	3	3	1	7	3	2
Total	3	5	6	2	4	4	3	4	2	7	5	2

Chi-squared tests revealed no differences between instructional conditions, between novices and experts, or between males and females. McNemar tests of pre- and post-test understanding showed that both novices and experts improved over time ($p < 0.0039$ and $p < 0.0025$), that both males and females improved ($p < 0.0078$ and $p < 0.0313$), but that in terms of the effect of instructional condition, only the subjects in the syntactic condition improved from pre- to post-test ($p < 0.0313$).

Modal systems

The number of Modal systems subjects understood was noted at pre- and post-test. This analysis was based on two questions, one which asked for explanations of Modal systems, and the operator reduction question described above.

Since three systems were described in the material, scores range from 0 to 3. Table 3.9 summarises the means and variances at pre- and post-test.

Table 3.9: Means and variances for the number of Modal systems understood

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Pre	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0.5 (0.75)	0 (0)	0 (0)	0 (0)	0.6 (1.44)	0 (0)
Post	3 (0)	0.6 (1.44)	2.83 (0.14)	2.5 (0.25)	0.75 (1.69)	0 (0)	1.33 (0.89)	1 (1.5)	1.5 (3.37)	1.14 (1.84)	2.2 (1.36)	1.5 (2.25)

Because so few people were able to use any of the systems at pre-test, an $[3 \times 2 \times 2]$ ANOVA was carried out on the post-test scores alone. This revealed main effects of instructional condition ($F(44,2)=5.027$, $p < 0.012$) and gender ($F(45,1)=4.358$, $p < 0.044$) with a trend for an effect of expertise ($F(45,1)=3.51$, $p < 0.069$), with experts out-performing novices and males out-performing females.

Figure 3.8 shows the average number of systems understood at pre- and post-test. Tukey tests revealed that subjects in the syntactic instructional condition understood significantly more systems at post-test than their counterparts in the diagrammatic condition ($p < 0.05$). There were no significant differences between subjects in the syntactic and learning environment conditions, nor between subjects in the learning environment and diagrammatic conditions.

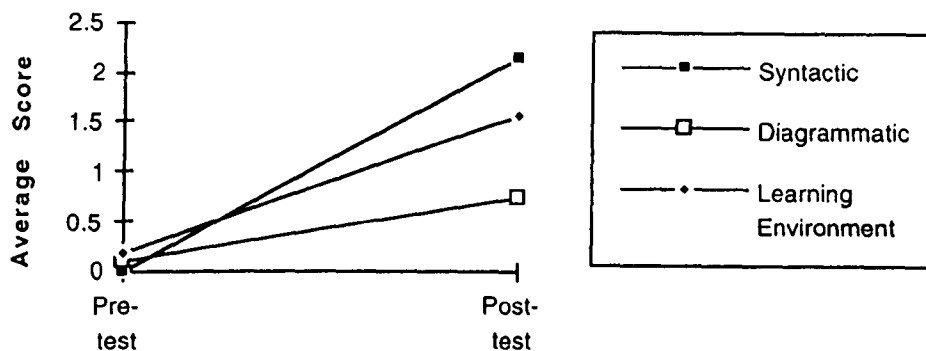


Figure 3.8: A comparison of the average number of Modal systems understood

Modal proof construction

Subjects were classified according to their ability to construct a Modal proof. Table 3.10 summarises the number of subjects who were able to construct proofs at pre- and post-test.

Table 3.10: Number of subjects who understood possible worlds

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Pre	0	0	0	0	0	0	1	0	0	0	1	0
Post	0	0	1	0	0	0	0	0	0	1	2	1
Total	3	5	6	2	4	4	3	4	2	7	5	2

Numbers were too small to permit statistical testing. However, it is interesting to note that the subject in the diagrammatic condition who was able to construct proofs at pre-test failed to demonstrate this ability at post-test. Five opportunities were given to demonstrate proof construction. Also interesting is that 80% of those able to construct proofs at post-test are from the learning environment condition.

McNemar tests show that neither expertise, gender nor instructional condition had any effect on improvement in ability to construct proofs.

Near-Transfer

Too few subjects attempted the near-transfer question to permit statistical testing. However, a summary of responses is presented in Table 3.11.

Table 3.11: responses to the post-test near-transfer question

	Syntax	Diagrammatic	Learning Environment
Attempted question	7	6	6
Correctly understood the new system	6	3	5
Constructed an adequate proof	5	1	3
Able to give a real-world example of when this system could occur	2	1	5
Total in condition	16	15	16

Although roughly equal numbers of subjects from each instructional condition attempted the near-transfer question, slightly more from the syntactic group were able to construct proofs, whilst more in the learning environment could give examples of when such systems might arise.

As with other measures of improvement, fewer subjects in the diagrammatic condition were able to complete the question.

3.5 Results: Post-test measures

Time

The length of time taken to complete the instructional material was noted. For simplicity, this was rounded to the nearest five minutes. Table 3.12 summarises the means and variances of the time required.

Table 3.12: Means and variances for time blocks taken

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Post	4.7 (6.2)	3.8 (3.76)	4.5 (3.25)	2.3 (2.25)	5.75 (8.19)	4 (1.5)	6 (6)	6.75 (8.19)	3.5 (6.25)	5.86 (8.12)	4 (3.2)	6 (1)

The time taken was analysed using an [3x2x2] ANOVA with between group factors of instructional condition, expertise and time. There were no main effects, and no significant interactions.

In order to investigate whether or not spending more time on the intervention material was beneficial, the time taken was correlated with measures of conceptual gain.

Study time was not found to be significantly correlated with the number of Modal systems subjects learnt, nor with measures of their post-test understanding of reference. Time was, however, found to be correlated with improvement in ability to apply Modal systems to operator reduction problems, although not in the direction expected (Spearman’s 1-tailed, $r=-0.3355$, $p<0.03$). Subjects who spent longer working through the instructional material were less likely to answer the operator reduction problems correctly.

Motivation

Subjects were asked to rate how much they enjoyed the study using a 5-point scale. Table 3.13 summarises the means and variances of the responses.

Table 3.13: Means and variances ratings of enjoyment

	Syntactic				Diagrammatic				Learning Environment			
	Novice		Expert		Novice		Expert		Novice		Expert	
	M	F	M	F	M	F	M	F	M	F	M	F
Post	3.3 (0.89)	2.6 (3.84)	3.8 (0.81)	3 (1)	3.25 (1.69)	4 (0)	4 (0)	3.5 (1.25)	4.5 (0.75)	3.29 (2.2)	4.2 (0.56)	5 (0)

Ratings of enjoyment were analysed using an [3x2x2] ANOVA with between group factors of instructional condition, expertise and gender. There were no main effects, and no significant interactions.

However, instructional condition played a role in whether subjects believed they had performed adequately or not (chi-squared, 2 d.f., $p<0.02$). Although few subjects who used the diagrammatic condition claimed to dislike it, all but one of them thought they had performed badly. By comparison, 56% of the subjects in the learning environment condition believed that they had performed “o.k.” or better, as did 37.5% of the subjects in the syntactic condition.

It should be noted that subjects' self-perceived improvement was not entirely accurate. Inspection of the data revealed that only subjects whose scores worsened accurately assessed their performance. Equal numbers of subjects whose scores improved thought they did badly as thought they did well. A chi-squared test confirmed that subjects whose scores improved or stayed the same were less able to gauge their performance than subjects whose scores worsened (d.f. 1, $p < 0.005$). A similar trend has been seen in other empirical studies involving subjects learning about abstract, formal concepts (e.g. Fung, 1988).

Previous studies (e.g. Fung et al. 1993; 1996) suggest that motivation is linked with performance on measures of logical ability. To see if this finding was replicated in this study, subjects' self-assessed measures of confidence and enjoyment were correlated with performance on measures of ability.

Enjoyment was found to be positively correlated with post-test performance on the operator reduction question (Spearman's 1-tailed, $r = 0.1862$, $p < 0.029$), and with the number of systems understood at post-test (Spearman's 1-tailed, $r = 0.2742$, $p < 0.031$). Confidence in performance was also correlated with scores on operator reduction (Spearman's 1-tailed, $r = 0.6545$, $p < 0.001$) and number of systems understood (Spearman's 1-tailed, $r = 0.5042$, $p < 0.001$). Confidence was also correlated with pre- to post-test improvement on the operator reduction question (Spearman's 1-tailed, $r = 0.4637$, $p < 0.001$), whilst enjoyment was not (Spearman's 1-tailed, $r = 0.2015$, $p < 0.087$). These results confirm a link between motivation and performance.

Whilst enjoyment was not positively correlated with the number of systems known at pre-test (Spearman's 1-tailed, $r = 0.1862$, $p < 0.105$), there was a trend for this measure of performance to be correlated with confidence (Spearman's 1-tailed, $r = 0.2152$, $p < 0.073$). However, this simply seems to indicate that subjects who had already studied Modal logic were sure they would perform well on the tests.

Representational preferences

As part of the post-test, subjects were shown a page typifying each of the other two instructional conditions. They were asked to compare this with the material they had used, and were invited to indicate which of the three representational styles they would have preferred to use. 21% either failed to respond or had no preference, but of the remaining subjects, 78% said that they would have preferred the learning environment material. A further 11% stated a preference for a combination of this with the syntactic condition. No-one from any of the three instructional conditions indicated a preference for the diagrammatic material. The responses of subjects who expressed an opinion are summarised in Figure 3.9.

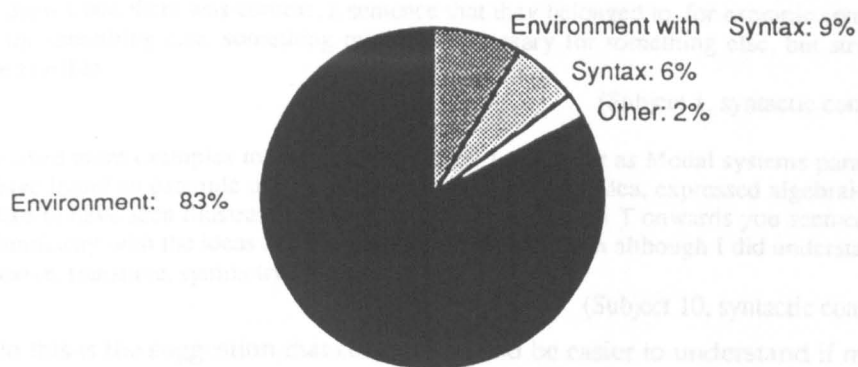


Figure 3.9: Expressed preferences for different representational styles.

3.6 Results: Qualitative data

Comments were gathered during the study, and analysed qualitatively. This analysis generated several categories, described below, which help explain the problems faced by students learning Modal logic. A full record of qualitative responses can be found in Appendix 3.

Not all topics in Modal logic are equally difficult

Subjects self-assessed problems suggest that of the seven concepts covered in the instructional material, some were far more difficult than others. Table 3.14 summarises the proportion of subjects who expressed difficulties with these concepts, and compares this with the proportion that the performance measures show actually had difficulties at post-test.

Table 3.14: Descriptive summary of conceptual problems

	Showed some understanding of the topic (post-test)	Believed they understood this topic (post-test/interview)
Use of Modal Operators	55%	51%
Possible Worlds	77%	Not asked
Reference	74%	77%
Belief and Modal logic	85%	Not asked
Modal systems	53%	28%
Modal proof construction	19%	15%
Accessibility relations	Not tested	53%

This shows that subjects found reference fairly easy, with the use of accessibility relations and Modal operators being moderately difficult, and Modal systems and proof construction appearing difficult to learners.

The only notable mismatch between performance and self-assessed knowledge was over the use of Modal systems, where results indicate that about twice as many people gained some degrees of conceptual understanding as believed they had.

Examples help concept acquisition and comprehension

A topic of clear importance, commented on by 51% of subjects, was that more examples would have been helpful. These comments cover both examples presented in the intervention material, and exercises to be completed by the subject. The lack of examples proved particularly difficult for subjects in the syntactic and diagrammatic conditions, with 43% and 45% (respectively) of responses coming from subjects in these instructional conditions.

Comments suggest that more examples would have make the concepts seem more meaningful:

I understood them when there was context, a sentence that they belonged to, for example something is necessary for something else, something might be necessary for something else, but strings of symbols were horrible.

(Subject 1, syntactic condition)

I would have liked more examples to illustrate meanings. I got as far as Modal systems para. 2 but then would have found an example useful. And there on every new idea, expressed algebraically, I would have like to have seen illustrated by an example. From system T onwards you seemed to be assuming a familiarity with the ideas and I couldn't follow very much although I did understand the concepts reflexive, transitive, symmetric.

(Subject 10, syntactic condition)

Closely related to this is the suggestion that concepts would be easier to understand if more examples were provided.

I think that by virtue of the fact that you've given me a conceptual explanation of what this is trying to do instead of something concrete to start with, that has locked my understanding. But if I had a concrete example first and then moved from that to this then I think that yes, I could follow the train of thought.

(Subject 27, diagrammatic condition)

These results give support to the arguments for using simple, concrete representations to provide content for the reasoning process, as described in section 2.5, and also stress the importance of the provision of examples by software tools, as described in section 2.3.

Exercises support knowledge acquisition

Just as provision of examples appears to support understanding, so provision of exercises was argued to support the acquisition of knowledge.

Felt that there was too much information to absorb - maybe doing some examples as I tried to learn the material would've helped. I looked at these questions and blanked.

(Subject 29, diagrammatic condition)

Although a series of exercises was provided in the intervention material, these were presented as completed examples. Comments such as this suggest that requiring subjects to complete exercises would complement the provision of examples in supporting knowledge acquisition.

Abstract notation is intimidating

Subjects expressed a clear preference for using the learning environment material. Comments explaining this opinion suggest that much of its appeal was that it appeared less intimidating than the abstract notation used by the syntactic and diagrammatic material.

I would prefer this model [Learning Environment], because visually, it appeals. I can unpack it. I can... I would make a reasonable go at that, other than that. Um... Why? Because visually, you actually can clearly see what is being instructed, what the aim objectives are, what it is you have to arrive at, or what this picture is telling you about. It's a lot clearer, it's more precise. You can tell instantly. I don't know; it's just not possible for me to look at that [Diagrammatic] and put any value, and read anything into it other than a mess of symbols.

(Subject 7, syntactic condition)

That one looks simpler. It's less intimidating.

(Subject 19, diagrammatic condition)

By contrast, the syntactic material was off-putting for subjects.

It was a frightening reminder of what degree level text books were like towards the end.

(Subject 1, syntactic condition)

Symbols are difficult to interpret

One of the most frequently encountered difficulties was the interpretation of symbols, in spite of the provision of a glossary sheet. 66% of respondents commented that they had experienced this problem.

Once it moved away from example sentences into symbols and equations I became totally lost.
(Subject 22, diagrammatic condition)

Just couldn't get to grips with the symbols at all - I didn't understand the abstract theoretical side. It made sense when talking about actual concepts, but this was too abstract - I was lost. Symbols etc. needed more explanation - with examples, e.g. from real world (or defined world).
(Subject 45, learning environment condition)

Analysis of the backgrounds of subjects showed that prior experience with logic significantly affected whether or not subjects found interpreting symbols to be a problem (Chi squared = 4.204, 1 d.f., $p < 0.05$). This result can be seen as providing an insight into the assertion that a lack of prior formal training will prove detrimental to students learning logic (cf. Fung & O'Shea, 1992).

Manipulation skills can be acquired without being understood

As has been identified, abstract notation can be difficult to interpret, and a lack of examples can make it hard to attach meanings to concepts. These themes are both related to a further category of comments, which suggests that rote skills can be developed without acquiring any appreciation of what these skills mean. Subjects found themselves unable to relate processes to meanings. Again, this seemed to be more of a problem for subjects in the syntactic and diagrammatic conditions.

I was OK as long as I just thought in symbols. As soon as I tried to think in sentences I naturally came unstuck.
(Subject 15, syntactic condition)

I didn't really understand what I was doing, really; it was just manipulating symbols in the end, and you get a bit lost in the recursive nature of it, so I'm not sure how I did.
(Subject 14, syntactic condition)

This reinforces arguments for the use of complementary representations, which stress the integration of content and process.

The diagrammatic condition is not pictorial

Comments indicated that the diagrammatic condition was no better than the syntactic at demonstrating the meaning behind the processes being taught.

I thought that the tree diagrams were fine, if you'd had enough concrete examples at that time to relate the abstract notation, because that's all they were, really, to the real world examples, then I could have understood.
(Subject 24, diagrammatic condition)

This certainly doesn't aid my understanding. It's not relevant to me, because it has no connection with anything that I've done before, so it's not a picture as far as I'm concerned, it's words with boxes round them. It's not a picture. A picture tells you something. So I would argue that this is not a pictorial representation, because it's not meaningful, there's no connection between the representation and the meaning.
(Subject 27, diagrammatic condition)

These comments help explain the problems faced by subjects in the syntactic and diagrammatic conditions. Given the difficulties associated with using abstract notation which have already

been identified, interpreting symbols and applying formal skills will be made more difficult for subjects in these instructional conditions.

The learning environment representation is more 'real' than the others

Several comments suggested that there was an hierarchy of abstraction in the representations that were being used. The simplest were the examples drawn from Reeves & Clarke (1990), which involved simple everyday occurrences expressed in natural language, such as asking whether all redcurrants had to be red. Next most concrete was the learning environment material, followed by the diagrammatic and syntactic conditions, both of which were seen as highly abstract by subjects.

Yes, this [the learning environment condition] is the most real of the lot. It's the easiest to imagine if you've got a non-scientific mind.

(Subject 2, syntactic condition)

It depends on what level of understanding you're at. I'm at the stage of knowledge acquisition; I think that this [Diagrammatic] is aimed for someone who can synthesize. I think it's a higher order skill to aid your understanding... I think that's a higher-order level of representation.

(Subject 27, diagrammatic condition)

These comments illustrate the 'dimension' of abstraction which has been used to classify representations (e.g. Friedman, 1993), as described in section 2.4. Given that another category of comments suggested that abstract styles of representation are intimidating, the comments about an hierarchy of abstraction help to explain the motivational problems associated with logic learning, and support the proposal that concrete content representations help students learning abstract topics.

In order to explore whether, as suggested, instructional conditions were better suited to novices or experts, a series of t-tests were carried out on the measures of performance on the operator reduction problems. These compared firstly the comparative scores of novices, and secondly, within each instructional condition, the post-test scores of novices and experts. It should be noted that the sample sizes for these tests were small (n from 4 to 8); consequently, results should be interpreted with caution.

The following results were found within each instructional condition:

- Syntactic condition: no difference between novices and experts (5 d.f., $p < 0.21$)
- Diagrammatic condition: no difference between novices and experts (7 d.f., $p < 0.79$)
- Learning Environment condition: a significant difference between experts and novices (5 d.f., $p < 0.02$)

No significant differences were found between groups. To summarise, the only significant difference found was within the learning environment condition, where "experts" (those who had completed a prior course in logic) performed better than novices. These results suggest that none of the instructional conditions is of particular benefit to novices, and that, contrary to 'subjects' opinions, the learning environment material was of more use to experts than novices.

Assessment favoured the syntactic and diagrammatic conditions

Not all subjects found the post-test equally easy to complete. In particular, subjects in the learning environment condition felt ill-prepared, and unable to express what they had learnt.

I guess you are using these diagrams with the brick walls and everything to make logic more accessible to students, but thought it's quite nice and I like it, sometimes there's no direct correspondence between the diagrams and the mathematical symbols and the mathematical relationships. So that's where I was lost. When I went back to the post-test, I had a problem with getting back to the logic, and this led me to revert back to my initial notions about the meaning of the sentence, not related to what I had read in between.

(Subject 33, learning environment condition)

One explanation for this difficulty is that the test questions were presented using standard syntactic formal notation. This had been selected to ensure consistency between instructional conditions. Both the syntactic and diagrammatic conditions make extensive use of this formal notation, and contain examples expressed using this style of representation. The learning environment condition, by contrast, avoids such symbols, relying almost exclusively on concrete sentences for examples. As a result, subjects in the learning environment who were not previously familiar with logical notation (i.e. those classified as "novices") could be expected to have unique problems completing the tests. These additional translation-related difficulties could explain the novice-expert difference in performance described above.

This explanation is supported by comments on the difficulty of translating between the learning environment representation and the other two systems.

I think [the learning environment representation] would probably be easier to comprehend. But I'm not sure that I'd then be able to extend it into any of the other notation. Erm... I suppose into your diagrams, yes, I could fit that into your diagrams. With a certain amount of effort, translation would be pretty straightforward. But to go to anything like that — [indicates syntactic] I wouldn't have thought so. Not without quite a lot of effort.

(Subject 19, diagrammatic condition)

Given this mismatch between the representations used in the material and the tests, the performance of subjects from the learning environment condition becomes still more worthy of note. This group managed to perform at least as well as subjects in both other instructional conditions on all measures of performance, in spite of having been unfamiliar with the notation being used.

Summary

An analysis of the qualitative data gathered from post-test responses has identified several themes of pedagogic importance. Central to these themes are the importance of providing concrete examples in order to support learning. The analysis suggested that provision of more examples and exercises would have helped subjects to learn more effectively. Minimising the use of abstract notation would have improved motivation by making the topic seem less intimidating, and would have removed the problems associated with understanding what abstract symbols and skills mean. This also helps explain the problems associated with the syntactic and diagrammatic representations, which were viewed as far more abstract than the learning environment system. Finally, a mismatch between the representations used in teaching and assessment was identified for subjects using the learning environment representation. This suggests that these subjects, who nevertheless performed at least as well as their counterparts on

all measures of performance, were hampered at post-test through a lack of familiarity with the notation being used.

3.7 Results: Analysis of the case study interview

One respondent (subject no. 38) agreed to take part in an in-depth interview, which was used to explore the issues raised in the study. The subject was selected on the grounds that she had a broadly typical background, had failed to make any pre- to post-test improvements, and appeared to encounter many of the problems common to other subjects. She had been using the learning environment material.

Analysis of the transcript of this interview identified several categories of comments describing difficulties she encountered with topics covered in the material. This section discusses those problems. The full transcript can be found in Appendix 4.

In addition to the standard questions given to subjects as part of the post-test, the observer worked through several problems, encouraging the subject to create her own representational scheme in order to understand the concepts involved. The representational scheme is also described, and comments relating to representation creation are discussed.

The Effect of Prior Knowledge

In spite of claiming no prior experience, the subject did in fact have some degree of familiarity with first order logic. However, whilst this might be expected to have been a benefit, the subject saw it as a hindrance.

I think I was introduced to logic in a way that was very tricky, and it... yeah, it was just a bit odd. Which was on this knowledge representation course... And we just mucked about with truth tables, and you know, symbols on a table, and it just drove me up the wall... I just think it's so removed, it's like playing games, it just irritates me. Probably because I don't understand it. But then I think, because I'm intelligent, I should be able to understand it. Perhaps it's how it's presented.

Clearly, prior experiences can set up negative expectations, and so contribute to the motivational problems some subjects have about learning logic.

Low motivation and perceived difficulty form a vicious circle

Perceiving the topic as difficult led to the subject developing a low opinion of her progress.

INTERVIEWER: I mean, you've got as far pretty much as the cancellation.
SUBJECT: Yeah. That's probably not very far, is it?
INTERVIEWER: It is actually. It's as far as most people get.

She also explained that one of the reasons she had been unable to complete the syntactic part of the near transfer question had been motivation.

SUBJECT: I think I'm a bit scared of this sort of thing.

It became apparent that these elements were related. The harder the subject found the topic, the less motivated she became. With a drop in motivation, progress slowed, making the topic seem still more difficult. This led to a failure to attempt several post-test questions, in spite of the subject having understood the concepts involved.

Understanding Possible Worlds

Possible worlds caused concern, as their relevance was not obvious.

Possible worlds really annoyed me. It's like it's... um... it's like these forms of logic. I don't know... sort of re-inventing things, making assumptions. Like the idea of a possible world. I'm probably going off track here. But if you bring in the idea of a possible world, you could be here forever, saying, "well, in a possible world, we can assume... this and this and this." And I can't... I can see... I suppose what I'm very confused about is the use of this.

Provision of examples, explaining the role of this concept, helped to clarify it.

Later, when talking about how to classify possible worlds, she explained part of her concern.

I just view a possible world as so rich I can't break it down.

The effect of this was to make any analysis based on classifications of possible worlds pointless.

The problem was eased by introducing the blocks-world metaphor.

I mean, I could probably understand it more for a grid world.

The examples used in the learning environment material had been chosen with this concern in mind. Representations which are used to support formal reasoning need to achieve a balance between being able to express enough cases to make examples non-trivial, and expressing so many as to overwhelm users. This balance is referred to as an appropriate level of expressivity (Cox, Stenning & Oberlander, 1994).

The blocks-world metaphor used by the learning environment condition helped this student overcome the anxieties raised by the abstract concepts. When shown the other two styles of presentation, her expressed preference was for the 'blocks-world'-based learning environment.

SUBJECT:	I think I'd have liked my one actually.
INTERVIEWER:	Right. Why?
SUBJECT:	It's got pictures in it! You know, it's got those things.
INTERVIEWER:	Those things? The grid worlds?
SUBJECT:	...they obviously helped a little bit.

This result is similar to the findings relating to the use of concrete examples outlined in section 3.6. However, instead of reiterating that such examples are of benefit to learners, it identifies a proviso: realistic examples can be too detailed to be of benefit. These comments suggest that simplicity and the avoidance of irrelevant detail are important qualities for content representations.

Problems with Reference

The concept of reference caused problems, although most of these stemmed from a misunderstanding.

Ok, but that's... they're merely hyp... they're hypothetical. I can't see why that one... Oh dear. 'Cause it says here, $2 \times 3 = 6$ no matter what happens in the world. But you could imagine a world in which 2×3 equals something else. [Interviewer prompts for an example] Well, the multiplication symbol may stand for something else. [Interviewer points out an explanatory passage in the teaching material] Sorry! Right... Sorry. Yeah; but... they're meanings. But so is this. The symbols change their meaning but our interpretation of Mickey Mouse and Michael Jackson could change.

The key difficulty was that names and symbols were seen as qualitatively different. Symbols "stand" for something, which is seen to be an arbitrary relationship, whilst the names "mean" something — here, the reference is more closely attached to the referent.

This is a common misconception, shared by many subjects. 22 subjects only understood the problem of reference either for names or for symbols by the post-test; a further 12 showed no awareness of this concept at all.

Interpretation requires Context

Several of the problems identified by the subject related to a lack of context from which to interpret concepts. Examples need to be placed so that they can illustrate the effects of new assumptions as they are introduced.

It's the type of logics you describe, they make a lot of assumptions, and I suppose you'd have to get used to those assumptions and do some exercises and use them before you learn the next form of logic.

The subject identified abstraction as a key problem here.

It was so abstract, you see. I'm not saying I can't understand abstract things, but when it's a new area...

Asked how this problem could be dealt with, the subject replied,

How could it be made less abstract? I don't know actually. Because when it was talking for example about Mickey Mouse and, erm, you... usually, you think, to make something less abstract you use concrete examples, you use familiar examples, so when it was talking about Mickey Mouse and Michael Jackson, my brain still got in a twist. I don't think it makes much difference at all. [laughs] Total twist. You could have been talking about A and B, it would still have got in a twist.

The conceptual clash here was between the problem of reference and the need to provide a context. When the subject created her own examples (see below), the observer prompted a solution to this, which was to explicitly state that names of objects could not change. This "concrete" gird world was then able to provide the context the subject required without introducing further complications.

Interpreting symbols

As already noted, previous experience with logic, represented syntactically, proved demotivating for the subject. This lack of motivation was reinforced by the difficulties she faced interpreting symbols such as stacked Modal operators. These are hard to express in natural language, but are common in Modal proofs, and understanding what they mean is an important part of learning Modal logic.

I could look up that, and then I could... there were some rules here, something which reminded me of predicate... something I'd come across. But there were two of them. So I didn't know what to do. So I was stuck. And there were two here.

Another subject had criticised the blocks-world metaphor, arguing that it did not adequately explain the Modal operators.

I'm unsure about the extent that the semantics of the accessibility relation really "captures" the meaning of the Modal operators.

(Subject 42, learning environment condition)

Indeed, the representation does simplify the operators, using the convention that each movement of a piece on the grid would introduce another layer of Modal operators. For example, if the ball is allowed to roll once on a grid, we could ask if (for some statement p) it was possible that p was true ($\Diamond p$). If we allowed it to roll twice, we could ask if it would be possible that p could become true after the second roll ($\Diamond\Diamond p$). Although difficult to express with natural language, this convention is easy to represent in terms of levels on a grid world diagram.

The subject found this simplification helpful.

INTERVIEWER: Can it be the case... that something can happen... So we're adding another layer onto that.

SUBJECT: Oh, right.

INTERVIEWER: That's what I'm trying to distinguish here. And in that case, no, it can't.

SUBJECT: Cause you've just said can it be... and I said yes, but I was thinking of it in another way...

INTERVIEWER: Yes, and that's exactly the point. How you interpret "can"; what's going to hold.

SUBJECT: Ah...

INTERVIEWER: So the conditions you're allowing to happen affect what's going on. Strictly, on the accessibility relation you've done, what I'm saying is, is it possible that "(possible (possible p)) is not true, but that possible p is true". Now if you change the relationship so that it doesn't have to move...

SUBJECT: So that describes the "can it be (can it be)"... Ahhh... I'm getting this. Yes.

Interpreting negatives

The subject had particular problems when dealing with negative propositions. These were made easier simply by altering the wording.

INTERVIEWER: Can it be the case that these things must not be in a straight line.

SUBJECT: I can never understand negatives like that. Can it be the case that...

INTERVIEWER: That these things aren't in a straight line. Ok, let's put it another way. Can it be the case that nothing is in the middle square. Well, not on any of these. So on this one... let's write this down. [writes it] Well, what's the next step? Can it be the case that it can be the case that nothing's in the middle square?

SUBJECT: Yes... loads!

The subject's inability to transform double negatives into positives formed a stumbling block which prevented certain questions from being attempted. This skill, although fairly minor in itself, can clearly have an important effect on students' attempts to learn logic.

Self-explanation and visualisation helped identify areas of conceptual difficulty

Some key concepts remained problematic until the subject attempted to explain and graphically represent what she was describing. For this student, visualisation through graphical representation formed an integral part of self-explanation.

I thought that was a big problem with possible worlds [i.e. infinite amount] [draws a note] This is an infinite number... I'm very visual... of possible worlds. And they could relate in a number of ways. Oh; right. Yeah, I get that. All this... all that Modal logic does, is that it imposes a structure on this set of worlds, by saying this is linked to that, or... it could be linked in another way.

However, it was not always helpful to use the examples provided in the material.

I'm using these as examples and it's throwing me.

Consequently, the interviewer and subject agreed to create a new system of possible worlds, defined by the subject. The subject decided that they would continue to use a blocks-world metaphor, as they had become used to this, and had seen the problems that arise from representations with an inappropriate degree of expressivity.

SUBJECT: Well, what shall I describe in my grid world?

INTERVIEWER: Whatever you want. You don't have to have it as big a grid as that if you want to keep it simple. Just do it like a noughts and crosses grid.

SUBJECT: Can't even do a noughts and crosses now, I'm thinking too hard! Like that?

INTERVIEWER: Yes.

SUBJECT: Ok, this is not a possible world.

INTERVIEWER: Why not?

SUBJECT: Ok, this is a possible world.

INTERVIEWER: Why?

SUBJECT: We can say that this is a possible world.

INTERVIEWER: Ok, that's good enough.

SUBJECT: Right, so that's a possible world, but there might be other ones.

After deciding on a framework for the grid, the subject decided what could be done with it.

SUBJECT: That's a circle, because I've got a bias here. And we've also got... no, I don't want to have it noughts and crosses. We've got a circle, and a triangle... can I have one more?

INTERVIEWER: You can have whatever you like!

SUBJECT: And a diamond.

INTERVIEWER: So you've got three shapes and your possible world. Ok, so what can your shapes do?

SUBJECT: They can move.

INTERVIEWER: What, all of them?

SUBJECT: Yep. They can all move.

INTERVIEWER: How far can they move, and in which directions?

SUBJECT: Well, each one... [laughs] can move... I'll look at the constraints of the space in a minute, I'll just say what the objects can do, first. There's going to be constraints in the possible world. Is that allowed, too?

INTERVIEWER: You can allow whatever you like! This is you making up a world.

SUBJECT: Oh, right. So each object can move up, down, left and right. This is very simple.

INTERVIEWER: How far can they move?

SUBJECT: Oh, just one. One box.

INTERVIEWER: Right.

SUBJECT: Now there's already a constraint, because if they're right by the edge, they can't move out. That's fair enough.

INTERVIEWER: And you can't add any in, or take any away?

SUBJECT: Nope. They have to stay there. Is that describing accessibility relations?

INTERVIEWER: Yes.

In terms of complementary representations, this process led to the development of a new content representation. These worlds were then combined into Modal structures using the same box-and-line process representation as that used in the diagrammatic and learning environment condition.

SUBJECT: Yeah, that's better. Now what?

INTERVIEWER: Well, let's see where you can get to from that. How many of these; how many new possible worlds can you get to?

SUBJECT: Loads!

INTERVIEWER: How many? Don't say loads!

SUBJECT: I don't know!

INTERVIEWER: Well, have a look. Now you've got three of these places that you can get to.

SUBJECT: Yes. That can go there, that can go there, or that can go there.

INTERVIEWER: Ok, then draw those.

SUBJECT: So you've got three possible... Oh; I'm drawing it in a box, but it's not a box. Sorry.

INTERVIEWER: It's like a cross shape. That's fine.

SUBJECT: It's on a piece of paper, that's all! So this... it could be that that happens, in which case it'll look like that, or... you don't need me to spell this out.

INTERVIEWER: Do it anyway.

SUBJECT: ...or that could happen, the circle moves to the centre, or the diamond moves. Yes?

The work-scratching which accompanies this exchange, showing an example of the grid and objects that were used, is shown in Figure 3.10.

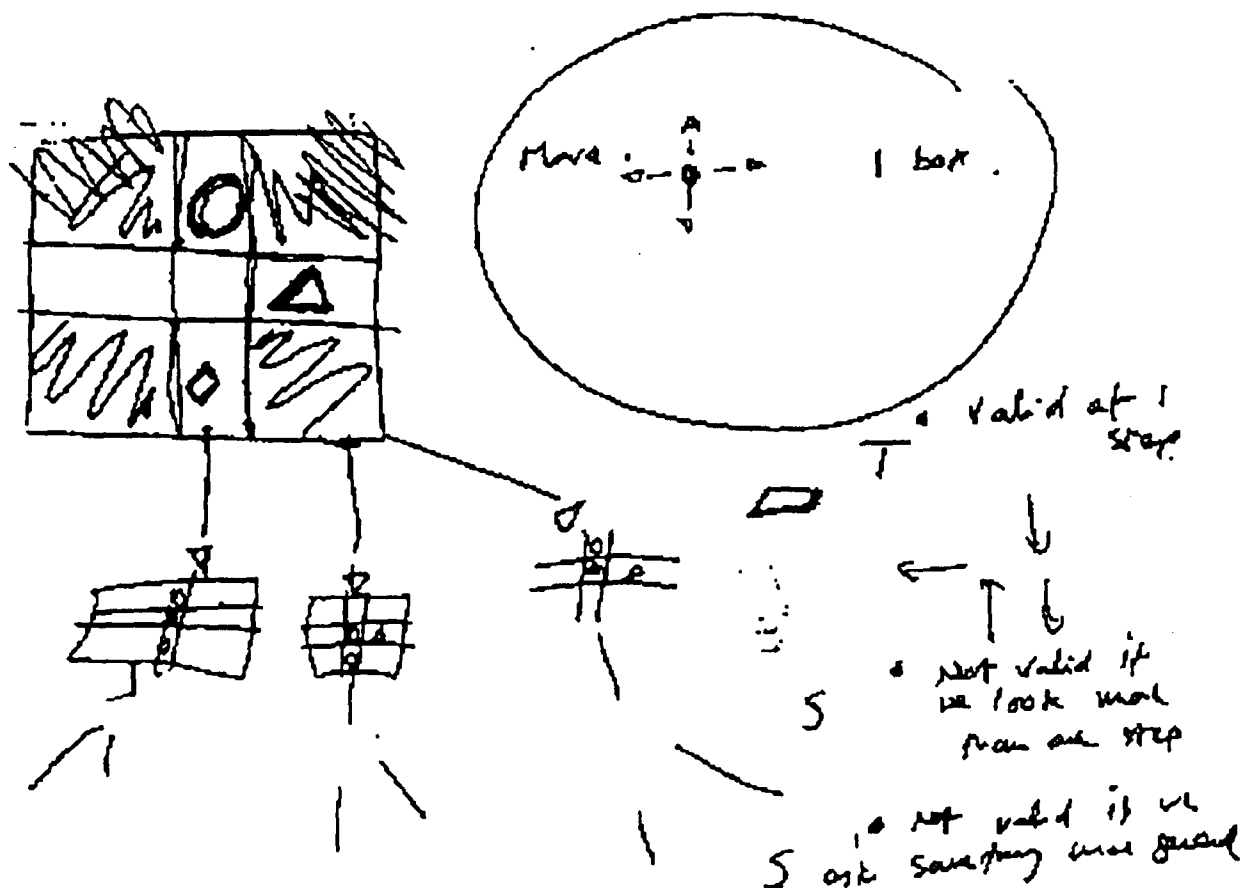


Figure 3.10: An example work-scratching showing the grid worlds created by the subject.

Questions were then devised which could draw the relevant concepts out from this example.

INTERVIEWER: So what can we say about this system?
 SUBJECT: I don't know.
 INTERVIEWER: Well, let's just think of anything. Um... let's think of how these shapes are related to each other, because that's an easy one, you can just read that off. Erm... can the circle be below the diamond?
 SUBJECT: Ah right, you can start predicting... This is to do with prediction, isn't it? Well... it could be in the future.

This representation was then used to work through several concepts, such as Modal systems, which had previously shown no improvement. They were found to benefit from explanations involving the subject's own system.

INTERVIEWER: What you're trying to do is, you're trying to prove whether this is a valid or an invalid statement, whether it's true or not. Ok? Would it be a valid conclusion to say that that's true, given this?
 SUBJECT: Ok.
 INTERVIEWER: And you've got this as your accessibility relation...
 SUBJECT: Right, am I just looking at this time, or...
 INTERVIEWER: Now, that's the question! What conditions must hold, given this, given that, to show whether that's valid or not.
 SUBJECT: I don't know what...
 INTERVIEWER: Well, you've just said it, really. "Are we looking at this step, or are we..."
 SUBJECT: Oh, right. Yes.
 INTERVIEWER: And you can see there's a difference coming out there, yes?
 SUBJECT: It holds here, but not here...
 INTERVIEWER: Ok. So write down that it's valid for one step, it's not valid if we look more than one step, and it's not valid if we just ask something more general, like, "can it ever happen that...".
 SUBJECT: Something more general...
 INTERVIEWER: That'll do. So, we've got a difference between this one and that one. There's the specific case, and we've got the more general cases. This specific one is T.

INTERVIEWER: That'll do. So, we've got a difference between this one and that one. There's the specific case, and we've got the more general cases. This specific one is T.
 SUBJECT: Aahh!!
 INTERVIEWER: Right. These more general ones, we'll split them up in a second, are the S systems.
 SUBJECT: That's what that is!

A second attempt at parts of the post-test demonstrated conceptual improvement, along with a 3 point improvement on question 3. The subject had learnt how to use system T, although she still had some problems with systems S4 and S5.

The subject was also able to attempt the near transfer question, successfully drawing the required diagram, although she was still unable to complete the syntactic proof.

It should be noted that although the comments indicate that the improvements resulted from working with a representation the subject had created and understood, it is impossible to rule out the possibility that these were caused by the individual support and extra study time that the interview provided.

The benefits of 'ownership' of a representation

Creating a new system of representation enabled the subject to develop a far better and more extensive appreciation of Modal logic. This process can be linked to the importance of providing examples and setting exercises for students; it makes the topic more meaningful, and provides practice for key skills and concepts.

In this case, although the subject was not clear about how much material she had understood, she was aware of having successfully learnt several key concepts by the end of the interview.

INTERVIEWER: Do you think that you learnt anything from that?
 SUBJECT: Oh yes! I did! I did!
 INTERVIEWER: Did you think that that was a useful way to go through it?
 SUBJECT: I did. I got a bit... and it might be because I'm saturated... because you do... I certainly understood all that.
 INTERVIEWER: All the different systems and how they work.
 SUBJECT: Yeah, basically.

Other subjects also called for the opportunity to create new systems, and stressed the importance of such a process for learning.

Like other people's diagrams — it/they reveals other people's thought processes — but doesn't make them mine. This is *alien* thinking and describing. 'Exposition' won't bridge the gap between us — nor will representation — I need an EXPERIENCE (a doing).

(Subject 41, learning environment condition)

Instructional material should support such exploration for all learners, encouraging the development of new systems with which to test their understanding of the topic.

Summary

This section has analysed an open interview with a subject from the learning environment condition in order to develop the grounded theory of Modal logic learning outlined in section 3.6.

Comments stressed the problems caused by the abstract nature of Modal logic. Blocks-worlds were identified as being useful in reducing this problem. The meaning and application of concepts were only acquired once they had been explored in the context of these worlds.

Creating a new system of representation also had a considerable impact on the learning process. The construction and exploration of this system proved extremely valuable as an aid to learning, enabling the subject to complete the post-test operator reduction questions which she had previously avoided. In terms of domain content, the process enabled her to master two of the three Modal systems (T and S4), and develop some appreciation of the third (S5).

A comparison with comments from other subjects shows that these problems were typical of those encountered in the study.

3.8 Discussion

Instructional condition

Reviewing the results in sections 3.4 to 3.7 shows that the diagrammatic condition proved extremely ill-suited to supporting learners. Contrary to expectations, both the syntactic and learning environment conditions were more effective on several measures of performance, including operator reduction problems and the number of Modal systems subjects understood. Clearly, the diagrammatic condition was less effective than the other instructional conditions at teaching the concepts involved in Modal logic.

Subjects in the syntactic and learning environment conditions performed equally well on most conceptual measures. Only two measures showed differences between subjects in these instructional conditions. The first difference was that four of the five subjects able to construct Modal proofs were from the learning environment condition. However, these numbers are obviously too small to generalise. The second difference is that only subjects in the syntactic condition showed significant improvement on the measure for understanding how belief interacts with Modal logic. Since the presentation of this part of the material was constant across all three sets of material, no effect of instructional condition was predicted. However, upon reflection, this prediction may have been unjustified, since the topic was presented syntactically, using natural language. Instead, it is possible that the change of emphasis in the diagrammatic and learning environment conditions towards less syntactic representations gave subjects in the syntactic condition an advantage on this topic.

Of the three instructional conditions, only the learning environment offered motivational benefits over either of the others. Not only were subjects significantly more confident about what they had learnt, but the learning environment style of representation proved immensely popular, being the style of choice for 89% of respondents.

Additionally, analysis of the qualitative data from the post-test and case study revealed several further advantages to the learning environment condition. It is perfectly possible to acquire manipulation skills without understanding what they mean or why they are used. Comments suggested that the provision of realistic, concrete examples supports this process of attaching meaning to skills. Of the three instructional conditions, only the learning environment representation was able to offer this support. The syntactic condition is wholly abstract; the diagrammatic condition was correctly identified as being just as formal, lacking any pictorial content. Comments clearly identified the learning environment representation as being the most

“real” of the three. They also confirmed that fewer subjects in the learning environment condition requested further examples to support learning than subjects in the other instructional conditions.

On a related theme, this concrete and realistic appearance contributed towards the motivational benefits of the instructional condition. Symbols and abstract notation intimidated subjects, leading them into a vicious circle of low motivation and hence greater difficulty.

The blocks-worlds used in this study were particularly well suited to supporting learning. The case study highlighted the problem of representations being “too real”, and overwhelming the user in a mass of irrelevant detail. By providing a simple yet believable situation, the learner was able to understand concepts that had previously proved to be too complex.

The other important point which was stressed by the case study was the importance of self-constructed systems of examples. These make the process all the more meaningful, helping learners to work through difficult concepts. The blocks-worlds formed a useful starting point for the development of such individual systems.

Expertise

Several measures showed an effect of expertise, defined in terms of prior experience with logic. Most important of these are that experts had less problems interpreting symbols and were better able to attempt the operator reduction problems than novices. These findings suggest that the role of expertise lies primarily with removing obstacles to learning, such as familiarity with notation, rather than with aiding conceptual development.

Time

The time required to work through the intervention material was not found to be affected by gender, experience or instructional condition. This ran counter to predictions, which suggested that the learning environment condition would take subjects longer to work through, and that the syntactic condition would be the fastest.

With few exceptions, scores on operator reduction problems worsened as subjects spent more time on the instructional material. Instead of reflecting deeper learning, it appears that the extra time required indicated that subjects were struggling. Subject’s comments give examples supporting this hypothesis, such as problems to do with the construction of diagrams in the diagrammatic condition.

The conditions under which you put asterisks above or below the line, for example, entertained me for some considerable time.

(Subject 19, diagrammatic condition)

The possibility that prior experience with logic might account for some of the variability (experienced subjects would not need to spend so long familiarising themselves with the symbols and procedures) was investigated, but no significant effect was found.

Motivation

In this study, a correlation between motivation and performance was found. This appears to support a finding from other studies (e.g. Fung & O'Shea, 1994), that the difficulty of the topic causes motivational problems for students.

However, comments gathered from subjects suggest that this relationship may not be quite so simple as it first appears. As demonstrated in the case study, due to a lack of confidence, some subjects failed to answer questions which they understood and were capable of completing. Here, poor motivation can be seen to cause 'difficulty', in the form of poor performance on measures of performance. As a result, this relationship must be viewed as a two-way one. This complex relationship clearly warrants further investigation.

Gender

Gender was found to play a significant role in only one conceptual or motivational measure: that of the number of Modal systems each subject understood. Additionally, two trends were observed. The first is that women are less likely to have any kind of familiarity with formal logic than men (Chi squared, d.f. = 1, $p < .10$). The second involved an interaction with instructional condition on the problem of reference, whereby the diagrammatic condition accentuated rather than moderated pre-test gender differences in favour of females.

Given that experience also affected the number of systems subjects learnt during the study, it seems possible that the link between gender and systems learnt is more consistent with an explanation based on prior experience of formal reasoning. Consequently, although gender was put forward as a possible factor influencing learning, these considerations suggest that it played only a minor role in this study.

Domain related results

In addition to comparing the effects of various factors on measures of performance, this study set out to investigate the problems associated with learning Modal logic. This investigation focused on seven key concepts: Modes, possible worlds, accessibility relations, Modal systems, Modal proofs, problems of reference, and issues about belief. Results showed that most subjects understood the problem of reference by the end of the study, and had few problems establishing the relationship between Modal operators and issues of belief. By contrast, Modal systems and constructing Modal proofs were found to be particularly difficult.

Most subjects were able to accurately assess how much they understood, the only notable exception to this being for the use of Modal systems. On this topic, almost twice as many people gained some degree of conceptual understanding as believed they had.

Quantitative results showed that only novices improved their performance on the question investigating understanding possible worlds. This suggests that the topic is particularly easy, and that the measure already suffers from a ceiling effect. Novices and experts, males and females alike showed significant improvements on the measure for understanding how belief interacts with Modal logic. It appears that this concept is also fairly simple to grasp, although

it should be noted that this study does not cover it in any great depth. Finally, the fact that no group significantly improved performance on proof construction confirms that this is a particularly hard skill to develop.

Analysis of case study data has provided a grounded theory of learning Modal logic. This is complemented by the data gathered from open questions on the post-tests. The theory stresses the importance of using simple concrete examples to support motivation, and to ground skills and concepts with meaning as they are acquired. It also identified how a vicious circle can develop between low motivation and the perceived difficulty of the topic. This re-emphasises the importance of attending to motivational issues when investigating how students learn formal topics.

Additionally, several domain-specific difficulties were identified. These include the realisation that symbols can change their meaning (the problem of reference), interpretation of negative propositions and double negatives, what a possible world 'is', and how accessibility relationships can make sense of an infinite set of possible worlds. The importance of the blocks-worlds in helping subjects understand the 'layers' of Modal diagrams was also made apparent.

The effectiveness of the learning environment representation

Work-scratchings were gathered from the work-in-progress notes of observed subjects, and from the pre- and post-tests of all subjects. Since no postal respondents returned their work-in-progress notes, it should be noted that this is unlikely to be a comprehensive collection. Appendix 5 collates the work-scratchings that were returned.

An analysis of the diagrammatic work-scratchings reveals that, of the 17 representations returned, only 41% included both relational information and details of the 'contents' of each world. Figure 3.11 shows an example concentrating on the law encoding, representational element of the representation; Figure 3.12 shows an example focusing on content.

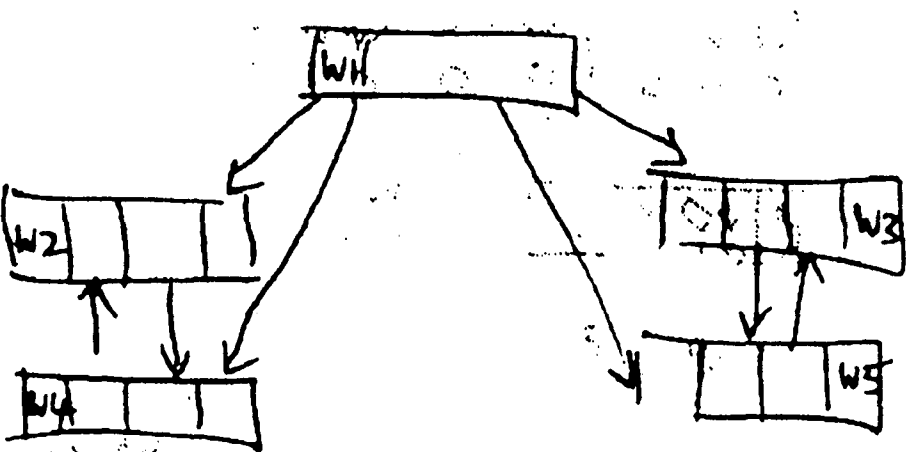


Figure 3.11: A work-scratching showing relational information only

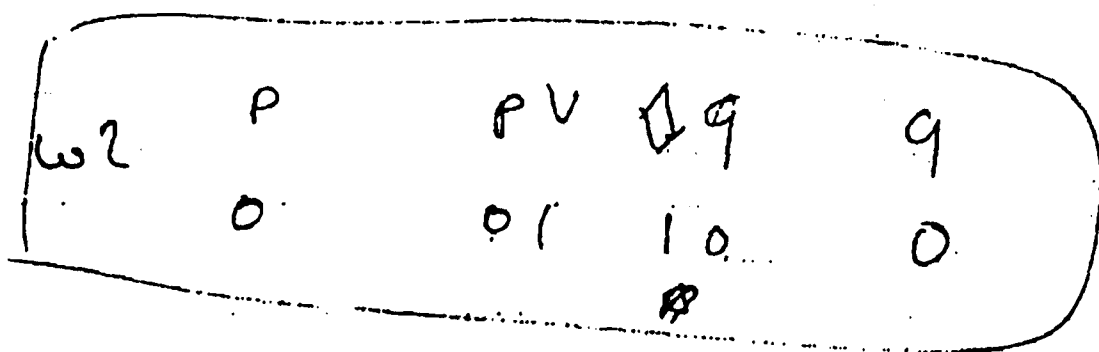


Figure 3.12: A work-scratching showing 'content' information only

Although these work-scratchings were gathered from both pre- and post-tests, and from any accompanying sheets of paper collected from the observed sessions, the majority were from post-test responses. Subjects were expected to provide complete solutions, incorporating both content and relational information. Very few work-in-progress diagrams contain both elements.

These data offer some insight into the uses made of complementary representations. The treatment of the process and content elements as separate entities suggests that the combination of the two in section 3.2 was inappropriate for learners. However, the even distribution of these between representations showing content and those showing process information suggests that both played an important role in supporting learning.

Summary

Instructional condition played an important role in concept acquisition. By contrast, the effects of experience were fairly moderate, and those of gender, minor. In addition to performing at least as well as the syntactic performance on all conceptual measures, subjects in the learning environment condition showed that this form of representation offered unique motivational benefits.

Time had a significant effect on scores, although not in the direction expected. Instead, improvement decreased as the time taken to work through the instructional material increased, suggesting that taking longer indicated that subjects were struggling.

The comparative performance of subjects in the learning environment and syntactic conditions (particularly in light of the mismatch of representations between teaching and testing materials), taken alongside the motivational differences between those two systems, suggests that the learning environment condition is the most effective of these three forms of representation for learners of Modal logic.

Problems associated with learning Modal logic were also investigated. These have led to the development of a rudimentary grounded theory for Modal logic learning. This stresses the importance of contextualisation, both on concept acquisition and skill development. Several topic-specific difficulties have also been identified.

Finally, an analysis of subject's work-scratchings showed that whilst both content and process representations played a useful role in supporting learning, they were rarely used in conjunction with each other.

3.9 A cognitive account of the performance differences between instructional conditions

Subjects' comments suggest one reason why the diagrammatic condition proved less effective than the others: a heavy cognitive load (Sweller, 1988) was involved in the construction of the diagrams. Each world drawn contains a formal proposition, which must be decomposed into its component truth values as part of the process of diagram construction. This was no trivial process.

I tried to work out, putting into words I could understand, whether p could be valid, and then, what that meant, and then what that meant [pointing to both parts of the biconditional], and trying split it into chunks, and then try to work it through.

(Subject 23, diagrammatic condition)

Shown below (Figure 3.13) is an example of subject 31's work-scratchings, attempting to decompose such a statement. This was the only record collected which shows the decomposition being worked through. As a result, we can conclude that all other attempts to construct these diagrams would have involved subjects performing this decomposition mentally.

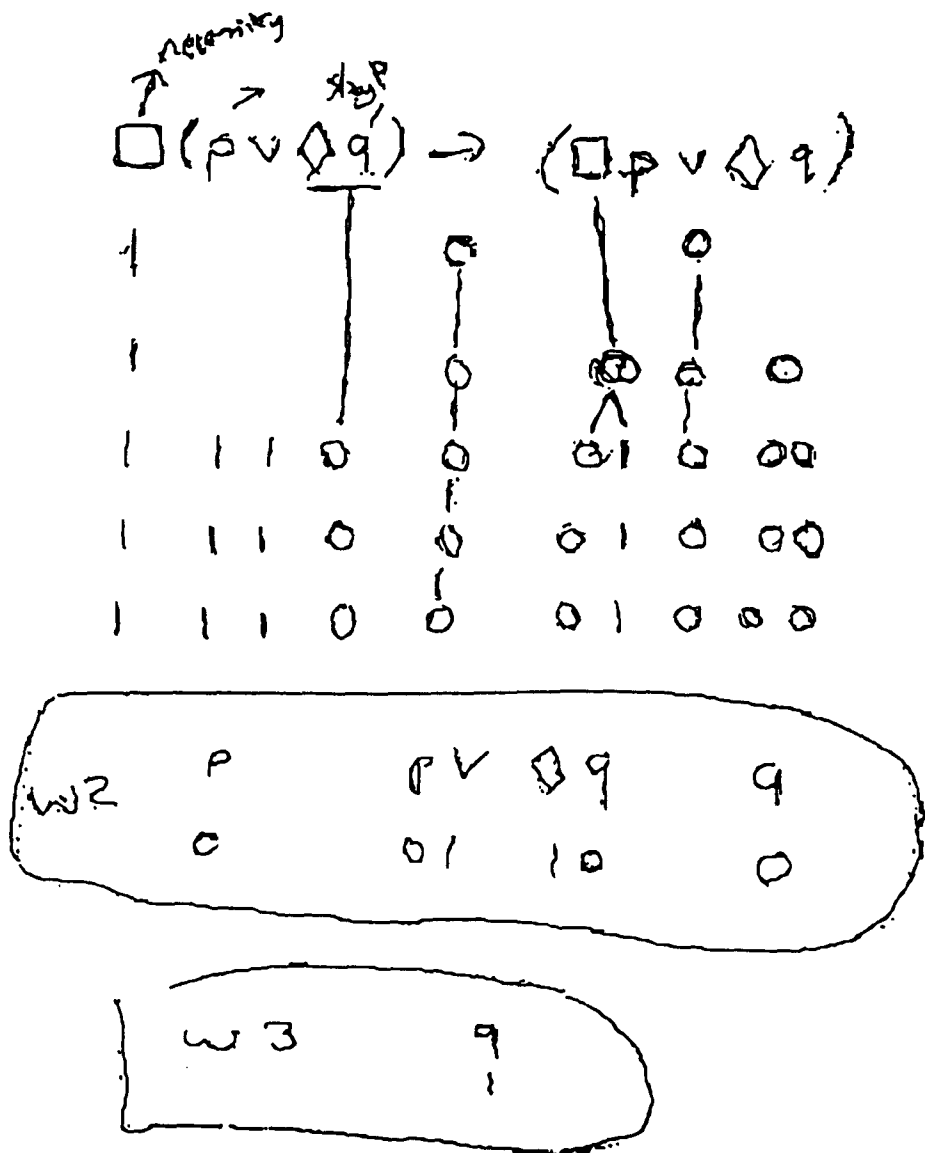


Figure 3.13: Work-scratching showing a decomposition of a statement

The process was particularly difficult for learners who lacked a background in first-order logic. Such subjects had to rely on the truth tables provided to look up each decomposition.

I think the problem is maybe because I don't have this background knowledge of basic logic at least, then it's difficult to put this information all together.

(Subject 25, diagrammatic condition)

The difficulties involved in constructing these diagrams diverted learners from concentrating on the concepts involved.

It depends on what level of understanding you're at... I feel that for those that are just looking at this to acquire knowledge, you're asking them to acquire a diagramming system which means nothing to them. But if they're synthesizing, they will have gone through acquiring the knowledge, and the understanding, and you know, being able to do all the other things. So that would mean something to them. I think that's a higher-order level of representation.

(Subject 27 diagrammatic condition)

Related difficulties can be seen in the comments of subjects in the syntactic condition.

I found it... not long winded, but I found it deep and difficult to follow. It would need to be unpacked a great deal, and put in more simplistic terms, to make sense of that.

(Subject 7, syntactic condition)

However, the load for subjects in the syntactic condition was considerably lighter than that of subjects using the diagrammatic representation. The construction of syntactic proofs involves the combination of simple true claims, rather than the decomposition of complex claims. Their problem lay in the initial interpretation of statements, a difficulty shared by subjects in all three instructional conditions, as mentioned in section 3.7.

I found it difficult to read the symbols. I had to keep looking back at the definitions. I couldn't find a way of 'saying' what $\Box\Diamond p$ meant.

(Subject 10, syntactic condition)

In addition to sharing the problems faced by the subjects in the syntactic condition, those in the learning environment condition had the additional load of remembering and applying the accessibility relation they were using. This was a far more complex relation than those in the other two instructional conditions, and created unique opportunities for confusion or misinterpretation.

Well, I thought the grid... was slightly distracting... For example, it was given at the beginning, for example the circle can't cross a wall, but it didn't play any part in the exercise or the understanding of the three systems, as far as I can see. I can see how that'd be useful in a further development giving more complex examples, but in terms of the examples we were dealing with... it's just that the wall added a further set of "what ifs..." in my mind that weren't relevant to what I was dealing with.

(Subject 44, learning environment condition)

Since all three instructional conditions shared the problem of interpreting symbols, the syntactic condition seems to have the lightest of the three cognitive loads. The detailed accessibility relation used in the learning environment condition added to this load. However, this extra burden is by no means as severe as that imposed by the decompositions in the diagrammatic condition.

This goes some way towards explaining the comparative performance of the three instructional conditions. Further insight can be gained from analysing the representations in terms of three of Green's cognitive dimensions (1989; 1994). Cognitive dimensions characterise the way that information artifacts structure and represent information; these include qualities such as hidden

dependencies, viscosity, abstraction level, premature commitment, secondary notation and diffuseness.

Viscosity

Altering the claim which is being proved has a surprisingly large effect in the syntactic condition. Although strings of characters in syntactic representations usually have a low level of viscosity, proofs are more problematic, since the alteration of one assumption can affect each subsequent line. In addition, changing the Modal system which is used can have far-reaching effects, as inferences valid in one system are not necessarily valid in another. System changes, therefore, can be almost as viscous as choosing a new claim to investigate.

The diagrammatic condition is even more viscous, since the alteration of one truth value in the starting world typically requires a completely new diagram. Changing Modal systems is less problematic, however. To change from T to S4 or S4 to S5 simply requires additional arrows to be drawn. Changes in the other direction are harder to deal with as they will often require the diagram to be extended.

The learning environment condition is more tractable. Unlike the other two instructional conditions, the representations are not tied to specific proofs. Instead, they describe examples for which a variety of claims can be proved by inspection. As a consequence, changes to the claim have little or no effect on the representation. The effect of changing Modal systems remains the same as for the diagrammatic condition, with certain alterations being simple, and others requiring the diagram to be extended.

Premature Commitment

The syntactic condition manages to avoid difficulties with premature commitment. This is because proofs involve assuming the truth of your assumptions and proceeding until either the proof is successfully completed or a contradiction arises. The learning environment condition also avoids premature commitment during diagram construction. Starting from a given situation, this process involves applying the accessibility relation to each world in turn. The diagram is complete when it has been extended far enough to demonstrate the claim or provide a counter-example. The truth of the claim is left unspecified until the diagram is completed.

By contrast, the diagrammatic condition forces subjects to guess a truth value at the outset. The rules for extending the diagram cannot be applied until a truth value is chosen and the statement decomposed. Unlike the syntactic condition, assuming the truth of claims is not standard practice, since it is often easier to prove a counter-example. Consequently, the degree of premature commitment when using the diagrammatic representation is far higher than of the syntactic or learning environment representations.

Abstraction level

The learning environment condition works by encoding a specific example, about which a variety of claims can be made. Although this variety indicates that the representation has a moderate

level of expressivity, the fact that the diagram encodes a specific example means that it possesses a very low level of abstraction.

By contrast, the syntactic and diagrammatic representations of Modal logic encode proofs. The effect of this is to allow the substitution of any equivalent statements for the symbols (such as 'p') into the proofs. As a result, these representations have a high level of abstraction. Furthermore, because both encode the same laws, their levels of abstraction are identical. Comments have already shown that subjects had difficulty dealing with this degree of abstraction in addition to learning the concepts which were being introduced.

Summary

A cognitive account of the representations seems to explain the unexpected results of this study. By this account, the initial hypothesis that subjects in the learning environment condition would out-perform those in the diagrammatic condition, and that these would out-perform subjects in the syntactic condition, was improbable. A better hypothesis would have been for subjects in the syntactic condition to out-perform those in the learning environment representation, and for those in turn to out-perform subjects in the diagrammatic condition. The learning environment's low degree of viscosity and lower level of abstraction, together with the problems of premature commitment in the diagrammatic condition, bring the re-ordering even closer to the observed outcome.

3.10 Conclusions

The initial hypotheses that the learning environment material would be more effective than the diagrammatic material, and that the diagrammatic material would be more effective for learners than the syntactic, was not found to hold. Instead, subjects using the syntactic and learning environment materials performed equally effectively, and both significantly out-performed those using the diagrammatic condition on important measures of improvement. Comments from subjects using the learning environment representation suggested that this occurred in spite of a mismatch between the representations used in the teaching and testing material.

Motivationally, the learning environment condition was more effective than the syntactic condition. This was due in no small part to its low level of abstraction and avoidance of symbols.

These two findings provide an answer to one of the research questions identified in section 1.3, about which uses of graphical representations best support formal reasoning. They also provide empirical support for the claims made in section 2.5 about the suitability of complementary representations for learners of Modal logic.

Responses to open questions, both from the post-test and from the case study, highlight additional benefits of the learning environment material. This material successfully provided examples of possible worlds rich enough to be used for non-trivial examples, but not too rich to make sense of. It also allowed subjects to create their own examples based on this representation, and to explore the constraints which define the systems, encouraging understanding rather than

just the memorisation of concepts and procedures. The stacking of Modal operators was introduced more gently through the use of 'layers' in the tree diagrams. These findings provide a first step towards answering the research question identified in section 1.3 relating the to suitability of the blocks-world metaphor for systems of logic other than classical first-order.

Several problems arose, regardless of instructional condition. More examples were requested, and support for proof construction was also required. Symbols were also a problem — in the diagrammatic and syntactic styles, it was difficult to ground these with a meaning; in the learning environment, less exposure to symbols led to the representational mismatch identified in the tests. Support for learning and interpreting symbols needs to be provided. It also became apparent that the relationship between the difficulty of the topic and students' motivation is not a simple causal one; instead, each factor can influence the other.

These considerations form the core of a preliminary grounded theory of Modal logic learning, contributing to an answer for another of the research questions in section 1.3 by identifying problems associated with learning Modal logic. These findings are complemented by the identification of two topics which are particularly easy for learners: the concept of possible worlds, and appreciating how belief interacts with Modal logic.

Having identified that the learning environment representation is well suited to supporting students learning Modal logic, the next stage of this research is to investigate whether using this as the basis of a software tool offers any benefits above and beyond those offered by the representation. Theoretical concerns such as cognitive load theory (Sweller, 1988), or the identification of interactivity as a beneficial quality for content representations, suggest that it would. Chapter four describes the design and implementation of a software tool based on the learning environment representation, which attempts to incorporate the results identified in this study.

More generally, this study emphasises that not all diagrammatic representations are good for learners. In this case, the law-encoding representation used in the diagrammatic condition proved less effective than either the syntactic or learning environment conditions. An account of the representations in terms of cognitive load suggests that most of the difficulties encountered by subjects in the diagrammatic condition were caused by high levels of viscosity, premature commitment and abstraction. This suggests that although law encoding diagrams can be more effective at teaching abstract topics than other forms of representation (c.g. Cheng, 1996b), their benefits will be moderated by the cognitive load involved.

Chapter 4

The Design and Implementation of MoLE

4.1 Introduction

This chapter describes the design and implementation of a software tool, the Modal Learning Environment (MoLE). MoLE provides students with a directly manipulatable microworld, which is based on the learning environment representation described in section 2.2.

First, this chapter outlines the software design aims. Section 4.3 describes how the learning environment representation has been adapted for use, taking into account issues identified in chapter 3. This representation is enhanced so that it can be used as the microworld which forms the basis of the learning environment. Finally, the design and implementation of the tool are described.

4.2 Design Aims

Chapter 3 demonstrated that the learning environment representation, which makes use of complementary representations, provides effective support for learners of Modal logic.

Implementing the representation as a software tool will enhance it in at least four ways: it will add interactivity, constrain users' errors, offer explanatory examples, and automate rote tasks.

This section outlines the design aims for the software tool MoLE, describing these four benefits.

Interactivity

Although the learning environment's content representation is concrete and pictorial, paper-based representations are only able to support a low level of interactivity. The first advantage of implementing the representation as part of a software tool is that this will allow it to support greater interactivity, in the form of direct manipulation. A consideration of the criteria for effective content representations in section 2.5 (e.g. Scaife & Rogers, 1996) suggests that this will make an implementation of the system more effective for supporting learners.

Additionally, implementing the representation in this way brings it in line with the blocks-world used in Tarski's World. This has been shown to encourage exploration and support the process of interpreting, building and manipulating logical statements (Fung & O'Shea, 1994).

Constraining errors

Implementing the representation as part of a software tool will constrain errors in two ways. Firstly, automating diagram construction avoids slips during proof construction. Secondly, using the blocks-world metaphor and offering examples that support (or disprove) claims allows users to check that their proof makes sense and has been formed consistently.

Offering explanatory examples

Section 2.3's review of software indicated that offering examples was a commonly-used technique to support students learning logic. This technique is even more important for Modal logic, where the use of examples and counter-examples to prove or disprove claims is standard practice.

Implementing the learning environment representation will allow the program to identify appropriate examples, providing a contextualised explanation of the results it generates. This feature directly addresses issues raised in section 3.6, which noted that examples play an important role in developing an understanding of Modal concepts.

The benefits of this feature are also supported by cognitive load theory (Sweller, 1988). MoLE will, in effect, be able to generate worked examples of proofs from a given starting condition. It will also automatically draw attention to salient features by identifying grid worlds which support or disprove the claims under investigation. Using software in this way should enhance users' learning (Sweller, 1989).

Automation of rote tasks

The fourth advantage proposed for the software is that it will automate rote procedures such as repeatedly drawing block worlds. Section 2.3 shows that this is another technique widely adopted by software to support students learning logic.

It has been argued that supporting mechanistic processes in this way should encourage exploration, so that students become familiar with and so understand formal concepts (Fung & O'Shea, 1992). One explanation for this process is that automation will free cognitive resources which would otherwise have been allocated to these procedures (Sweller, 1988).

General considerations

The instructional material from the study in chapter three used representations to illustrate concepts and support subjects' understanding of the topic.

The same format is adopted for the software tool MoLE. The software will act as a resource which can be used to develop and explore examples; it is not intended to act as tutoring system. Traditional printed teaching materials will be used to introduce and explain concepts, whilst

MoLE provides a means of exploring how these operate, illustrates what they mean, and gives support and practice in the form of exercises.

4.3 Adapting the representation for use in the software

Chapter 3 demonstrated that the learning environment representation, which combined law encoding diagrams with simple graphical examples, was an effective way of teaching Modal logic. This representation incorporated both a law encoding diagram and a blocks-world content representation, both of which have been used effectively to support students learning abstract topics (e.g. Cheng, 1996d; Fung & O'Shea, 1994). The decision to combine these representations was motivated by an analysis of the roles they played in formal reasoning, presented in section 2.5. This indicated that these representations serve complementary roles, with law encoding diagrams giving support to the process of reasoning, and learning environments providing content.

This section briefly re-describes the learning environment representation, and presents the changes motivated by findings from the study described in chapter 3.

The learning environment representation

The learning environment representation is comprised of two complementary elements: a series of grid worlds, which provide content for the reasoning process, and a tree diagram which encodes the laws of Modal relationships (Figure 4.1). The elements used by this representation are described fully in section 3.2.

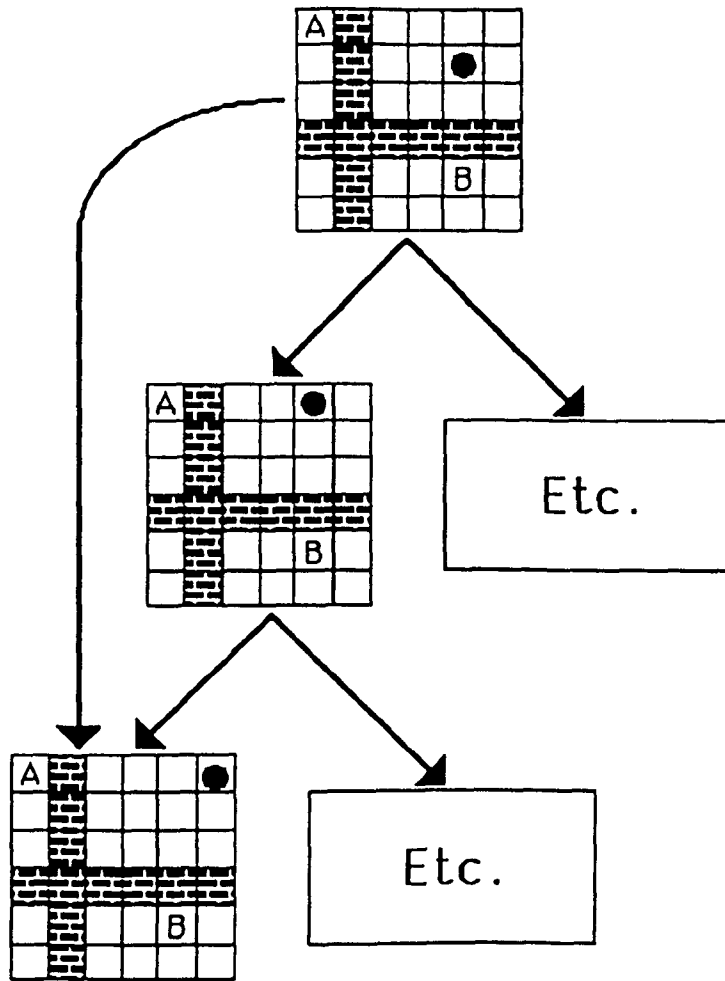


Figure 4.1: The learning environment representation, as described in section 3.2

The grid worlds contain three types of objects: spheres, which are allowed to move around the grid, walls, and objects (denoted by letters) that remain stationary. Relational statements are then used to form the basis of formal proofs. The “etc.” boxes are a convention introduced so that the existence of further grid worlds can be acknowledged without having to draw each of them.

The tree diagram consists of sets of arrows that connect grid worlds together. An arrow is drawn from one grid world to another if the second can be reached by applying the current accessibility relation to the first. If the relationship being used is transitive, further arrows can be added. Symmetric relations use double-headed arrows.

MoLE’s representation

The representation used in the implementation of MoLE differs from that presented above in several important respects. These are outlined below.

Separation of representations

The learning environment representation used in the study in chapter three (Figure 4.1) combines two complementary elements: a block world to provide content for the reasoning process, and a tree diagram to support the process of reasoning.

Two considerations prompted the separation of these elements when adapting the representation for use in MoLE. Firstly, showing all the grid worlds at every level of the diagram rapidly becomes impractical due to restrictions of screen size. As a consequence, only portions of the

diagram would be visible at any one time. Separating the block worlds from the tree diagrams allows an overview to be retained, and focuses attention on the elements of the representation most salient for problem solving. Secondly, as identified in section 3.8, few subjects integrated these elements when working with pen and paper. This suggests that using the two elements separately will be closer to the natural style of working of the students using MoLE.

Simplification of objects

The grid in MoLE has been simplified. Subjects' comments about the original grid world indicated that the different types of elements (objects, balls and walls) led to confusion about how each functions. In MoLE, all elements are referred to as "pieces", and obey a simple set of rules. Pieces have a name and one of three shapes (sphere, tetrahedron, or cube). In addition, if the piece is a sphere, it is allowed to move by as much as one square up, down, left or right, so long as that movement ends in an empty square on the grid. The claims which are investigated are formed by comparing the positions of different pieces.

Restrictions on grid size

The size of the grid used in MoLE is restricted, with a maximum of 6x6 squares. Ideally, systems should be restricted so that they do not represent superfluous abstractions, because these will hamper performance (Cox, Stenning & Oberlander, 1994). Strictly, this relates to the benefits for the user of diagrams which are unable to represent as much abstraction as syntactic representations (Stenning, 1993); however, it also holds true pragmatically for the number of disjunctions that need to be represented at each level of the tree. Unless considerable restrictions are introduced, the number of worlds created by each movement will increase exponentially, creating problems of orientation for users.

Restrictions on movement

The final alteration to the grid world involved restricting movement to spheres. This limits the number of worlds created by each movement still further, although greater expressivity can be achieved by de-selecting this option in MoLE. Typically, questions will only include one sphere, allowing a maximum of five new worlds to be created by each movement: one for each of the four directions (up, down, left and right), and one for the identity relation or null condition in which the piece remains stationary. In practice, however, constraints such as the positions of other pieces and the edges of the grid reduce this range still further. This convention seems to strike a balance between simplicity and diversity, providing an appropriate degree of expressivity (Stenning & Oberlander, 1995); however, empirical work will be required to confirm the usability of this system.

Simplification of links

The tree diagram was also revised. Rather than introduce special rules (such as double-headed arrows or the additional transitivity arrows) for different Modal systems, MoLE's diagrams are constructed simply with lines from each grid world to its immediate 'offspring'. More

experienced users, with an understanding of Modal systems, can then edit these diagrams to add features such as transitional links if they so wish.

Summary

Figures 2 and 3 give an example of a Modal diagram using the revised grid world system, as outlined above. Note that although the examples under consideration have been revised, the tree structure remains law encoding.

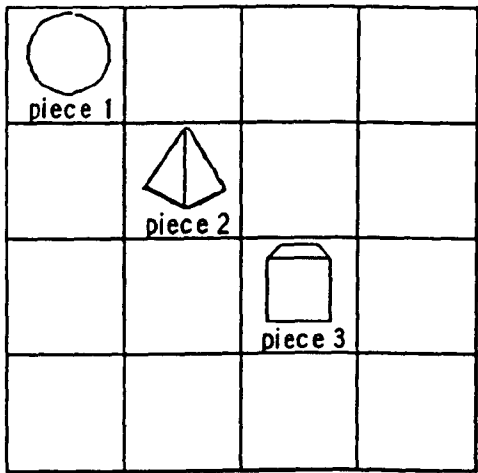


Figure 4.2: A grid world using the revised system

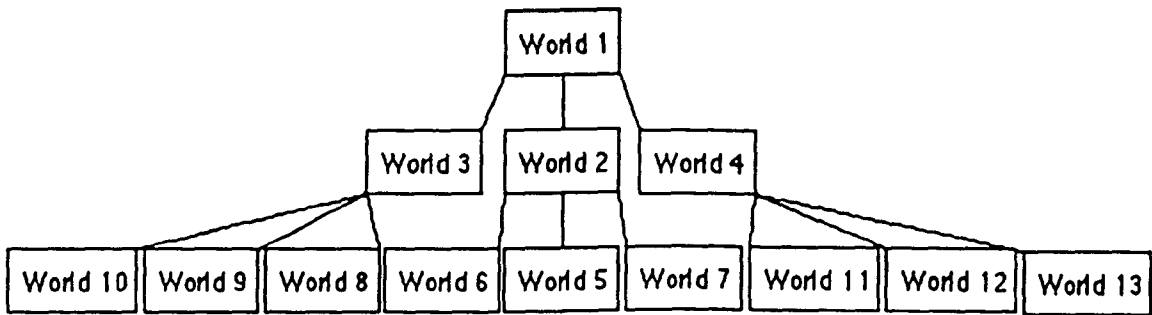


Figure 4.3: a Modal diagram using the revised system.

4.4 Software Design

This section describes the design of the software tool MoLE. An example of the kind of problem MoLE will be used to solve is presented, and the elements required to support this are identified. These elements are then discussed, and the section ends by providing a flow of control diagram showing how they are interrelated.

Design requirements

The software tool, MoLE, aims to provide a learning environment in which students can explore concepts by constructing Modal diagrams and assessing the truth of claims. MoLE will be used to create a Modal diagram by iteratively moving pieces and creating new worlds. Relational claims can then be evaluated for each world, and Modal claims can be evaluated for the diagram.

For example, a student might start by creating a grid world as shown in Figure 4.4:

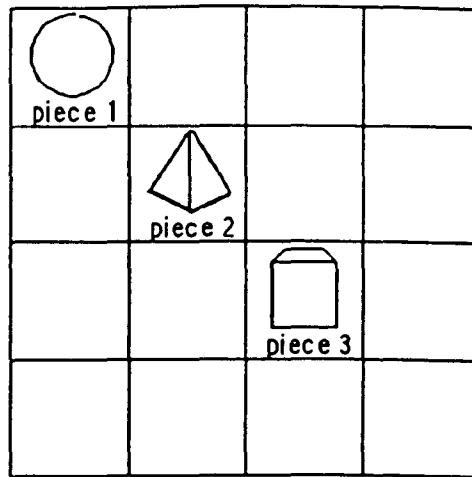


Figure 4.4: An example of an initial grid world.

By allowing spheres to move up to one square, three new worlds would be created. (One where the sphere moves right, one where it moves down, and one where it remains stationary) Figure 4.5 shows the resultant Modal diagram.

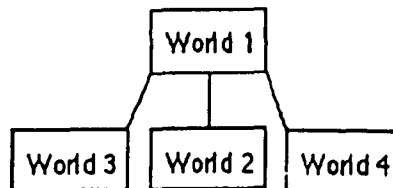


Figure 4.5: The modal diagram created by allowing spheres to move once

At this point, relational claims such as, “is piece 1 left of piece 2?” can be evaluated for each world, and Modal claims, such as “*must* piece 1 left of piece 2?” can be evaluated by inspecting the diagram.

In order to support this process, MoLE requires at least two elements: firstly, a model, which creates the pieces, grid worlds and diagrams, and secondly, an interface, which displays the information and allows users to manipulate and interrogate the model. The function of the interface could be further sub-divided into elements that display the model and elements that allow users to act on the model.

For this reason, VisualWorks 2.5 (an implementation of SmallTalk) was chosen as an appropriate programming language for the implementation. SmallTalk’s Model-View-Controller paradigm (Krasner & Pope, 1988) is particularly well suited to this kind of implementation because it already contains objects corresponding to each of the three elements required. This paradigm splits applications into three elements:

- The Model, which contains the information displayed by the application
- The View, which displays the model’s information, usually graphically
- The Controller, which manages the interaction between the interface and the model

Applications are built by creating subclasses of the appropriate objects. The remainder of this section outlines the design of the objects required for MoLE.

(i) The Model

The model for MoLE is built from three types of objects, organised hierarchically. The first and simplest of these is the Piece. Pieces only need to have a shape and a name.

The next object in the hierarchy is the Grid World. Essentially, this is a two-dimensional array in which pieces are placed. However, grid worlds can also be interrogated about the spatial relationships of the pieces. This allows users to find out, for example, if one piece is left of another. Grid worlds are also named, and keep track of other worlds to which they are connected. Normally, World A will be linked to World B if B is one of the worlds created when the pieces in world A are allowed to move.

In order to introduce Modal semantics, grid worlds are grouped together in a Universe. Although the grid worlds are responsible for keeping track of the worlds to which they are connected, it is the universe model which deals with Modal queries, and which interprets these links. The universe deals with queries by interrogating grid worlds to find the truth values of claims, identifying the other worlds to which the first is related, interrogating each of these about a simpler claim, and so on, until the query is answered. The process by which the related worlds are selected is governed by the Modal system in use; consequently, the universe also needs to record which system the user has chosen.

This hierarchy uses the dependency mechanism of SmallTalk. Universes interrogate and manipulate grid worlds, which in turn interrogate and manipulate pieces. In order to update the model, grid worlds become dependents of pieces, and universes become dependents of grid worlds. This means that any changes are passed back through the model, from pieces to grid worlds and finally to the universe, propagating further changes as necessary.

(ii) The View

VisualWorks Views are created by “painting” interfaces using a blank canvas and a menu of features, and then connecting this to the underlying model via a set of intermediate objects.

Many of the features required, such as lists, have View classes defined for them already. However, MoLE required two additional View classes: one for grid worlds, and one for universes. The GridView creates the representation of the pieces in each world, placed on a grid. The UniverseView displays the grid worlds, each of which is connected by lines to the grid worlds it is linked to. The UniverseView dynamically builds and revises the diagrams, providing a map-like overview of the grid worlds.

Pieces are only displayed in the context of a grid world; consequently, these have no need of a separate view class.

(iii) The Controller

As with views, VisualWorks already supports controllers for common features. However, MoLE requires four new controllers: one each for the GridView and the UniverseView, and one each for the rest of the functions in the Universe window and the Grid World window.

The Grid Controller supports direct manipulation by allowing users to place pieces and subsequently drag and drop them (within the grid) using the mouse. Additionally, double clicking on a piece opens a dialogue window that allows users to edit its name and shape.

The Universe Controller acts in much the same way. Grid Worlds can be created, deleted and moved using the mouse. Double clicking on a world opens up a new window containing the appropriate Grid View so that the world can be inspected or edited.

As with the View classes, standard features such as buttons and lists already have controllers defined for them. These needed no re-implementation.

Object relationships and dependencies, and flow of control

Although the Model-View-Controller mechanism is broadly suitable for this implementation, the formalism described by Krasner & Pope (1988) was not entirely appropriate for the three-part model required. Applying this as described would result in a complex and cluttered relationship between the model and its associated views and controllers. An alternative is to extend this formalism (ibid.) by applying the standard Model-View-Controller approach to each of the three models in turn, and then linking these distinct elements. The reason this approach is described as an extension of the formalism is that in the special case where only one model is required, it becomes identical to Krasner & Pope's system, and so is able to support all the applications their original formulation supported. However, in addition, it is also suited to modeling cases such as the one described here, where multiple Model-View-Controller mechanisms are required to work together.

The three-part Model-View-Controller mechanism, together with the three-part hierarchy of the underlying model, results in the control structure illustrated in Figure 4.6. Note that in this case, no views or controllers are associated with Pieces, as these are only encountered in the context of Grid Worlds.

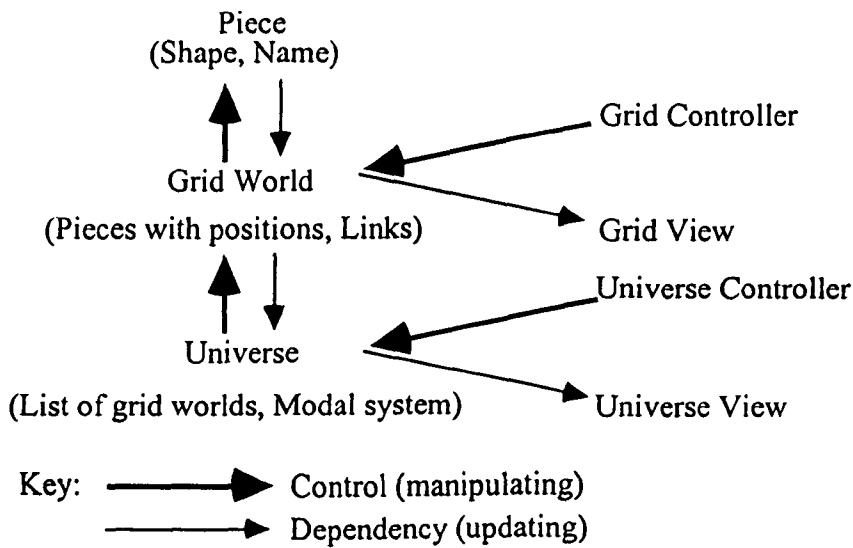


Figure 4.6: The relationship between the objects used in MoLE

4.5 Implementation

The design detailed above was implemented on a SUN Workstation using VisualWorks 2.5. This section describes an overview of the process of implementation. It starts by outlining the structure of a generic VisualWorks object, and summarises the objects required to support MoLE. The design modifications that were required are described, and the section concludes with a revised flow of control diagram which shows how these elements were integrated. The full code for MoLE can be found in Appendix 6.

A generic VisualWorks object

Objects

Objects in VisualWorks are created as instances of a Class. Each class is characterised by the variables it contains and the messages it responds to. Variables are usually given a descriptive name. VisualWorks requires all instance variables to start with a lower-case letter.

For example, the class *Fraction* has two instance variables, *numerator* and *denominator*, and a series of methods that describe how it should respond to a variety of arithmetic functions.

Classes

Classes are created as sub-classes of other objects. For example, the class of *Integers* is a sub-class of *Numbers*. All classes are sub-classes of *Object*, which provides the generic structure of *VisualWorks* objects. When sub-classes are created, they inherit the variables and methods of their superclass. The new classes can then add to or modify these methods as required.

In addition to an object's variables, *VisualWorks* permits what it calls "class variables". These can be accessed by any instance of the class, and irrespective of which instance accesses the variable, the same value will be returned. To illustrate this, a class called '*ClockFace*' could be designed which opens a window, showing the time on one of a selection of styles of clock face. If three or four of these windows were open at any one time, each instance of *ClockFace* could show the time using a different style of clock, but all would rely on a single class variable to hold the correct time.

Class variables are differentiated from instance variables by the first letter of their name. All instance variables start with a lower case letter; all class variables start with a capital. Because of this distinction, when sentences in this chapter start with an instance variable, the name will not be capitalised.

Interaction

Objects interact by message passing. When an object is sent a message, it returns an object in response. The type of object which is returned depends on the message being sent. For example, if the message "+ 3" was sent to the object "2" (an instance of the class *Integer*), the object returned would be "5" (another instance of the class *integer*). More complicated messages often involve the creation of a new object specifically designed to satisfy a task. For example, sending the object '2' the message 'negated' creates a new object (0-2) and returns this. When creating a new object in this way, the symbol '^' is used to indicate which object the message returns.

Polymorphism

VisualWorks supports polymorphism. This allows different objects to have methods that share the same name but which have distinct implementations from class to class.

One good example of this is the "PrintOn:" method. Every object will respond to the *PrintOn:* method; however, whilst some simply return a string describing their type (e.g. "a *ListView*"), others return a value (e.g. "true"), or a list of contents (e.g. "a list containing: ..."), depending on how the method has been implemented for that class of objects.

Accessing variables

An object's variables remain private to that object. In order to allow other objects to access those variables, two types of methods need to be created: these are referred to as *accessors* and

mutators. Typically, these are based on the name of the variable to be accessed, and take the following form:

```
aVariable
^aVariable
aVariable: aValue
aVariable := aValue.
```

The first of these, the accessor, returns the value of the variable, allowing other objects to “read” it. The second, the mutator, allows other objects to alter the value of the variable, so that other objects can “write” on it. By choosing to implement one or both of these, variables can be made that give read access, write access, or read-and-write access to other objects.

Summary

Objects in VisualWorks are created as instances of a class. Each class is defined as a refinement of some more generic class, and inherits the variables and methods of that superclass. These can then be added to and adapted, as required. New methods may include accessors and mutators, which allow other objects to access variables.

(i) Implementation of the Model

The design of the model required three types of objects to be created: pieces, grid worlds, and universes. This section outlines the implementation of each of these.

Piece

The class Piece was created as a sub-class of Object. It provided two new variables: shape and name. These were provided with accessor and mutator methods so that grid worlds could manipulate and read their values. It also provided a means of comparing pieces so that they could be sorted alphabetically when being displayed on the interface.

The GridWorld and Universe classes

The GridWorld and Universe classes were created as sub-classes of ‘Model’. This provided additional functionality which supported automatic updating using the dependency mechanism.

GridWorld: instance variables

The intended design of the class GridWorld included an array, which would be used to keep track of where each piece was placed. During implementation, it was decided to replace this with a dictionary, as this simplified the process of looking for pieces. VisualWorks dictionaries are sets of associations; each element has two parts: a key, and a value. In this case, the keys are pieces, and the values, co-ordinate positions for the grid. The program could simply access the association whose key was the piece being investigated and return the appropriate co-ordinate, instead of checking each square in turn to see whether or not it was empty, checking the name of any piece which it found, and then seeing if that corresponded to the piece which was being searched for. Searches by co-ordinate, which occur less frequently than searches by piece, were carried out by checking each association in turn to see if the value matched the co-ordinate, and returning the key of the association which satisfied the search.

The original design involved a distinct hierarchy of objects, with grid worlds only being able to pass information to universes through the dependency mechanism. During implementation, it was decided that a variable, 'universe', would be included that allowed grid worlds to access the universe they were contained in without using dependencies. This was to simplify three operations:

- The evaluation of claims. When claims are evaluated, the universe keeps track of the pieces being compared. Evaluation involves passing the claim along a series of connected worlds. It is easier for each to interrogate the universe separately than for additional variables to be created to identify the pieces.
- The creation of new worlds. When pieces are moved in a grid world, new worlds can be created. The universe variable allows the progenitor to add the new worlds to the universe.
- The naming of new worlds. When new worlds are created, the grid world names them by generating a suggestion and asking the universe whether it is already being used. The universe returns a truth value to indicate whether or not the name is acceptable. Whilst this value remains false, new names are suggested.

Instances of the class `GridWorld` have five variables: `name`, `connections`, `pieces`, `pieceToPlace`, and `universe`.

The variable 'name' has accessor and mutator methods, enabling the user to alter its value by typing in a text window in the interface. The variable 'connections' contains a list of the names of other grid worlds to which it is connected. 'pieces', as discussed above, is a dictionary of the pieces to be placed on the grid, together with their positions. 'pieceToPlace' is a Boolean variable which records whether or not the grid world includes a piece which has not yet been placed on the grid. This is used primarily when creating new pieces. 'universe', as discussed above, allows grid worlds to interrogate the universe they are a part of.

GridWorld: class variables

In addition to their own instance variables, grid worlds have eight class variables. These are: `CircleImage`, `SquareImage`, `TetrahedronImage`, `GridSize`, `OnlyCircles`, `System`, `HasWarned`, `Example`.

The first three class variables (`CircleImage`, `SquareImage` and `TetrahedronImage`) return the images used by the grid world's associated View to draw pieces. This approach was chosen because it improves the speed at which the display draws the representation, making animation smoother.

The next three class variables (`GridSize`, `OnlyCircles` and `System`) describe how the grid worlds behave. 'GridSize' contains a co-ordinate, used to fix the size of the grids. For example, if 'GridSize' was 2@3, the grids would be 2 squares long and 3 high. 'OnlyCircles' is a Boolean variable that records whether all pieces (or just circles) are allowed to move when new worlds are generated. 'System' keeps track of the Modal system (T, S4 or S5) being used.

The last two variables (HasWarned and Example) are to do with evaluation. When the worlds are parsed to evaluate a claim, two errors can occur: a world might not contain one of the pieces being considered, or the diagram might not contain enough layers to evaluate the claim.

'HasWarned' is set to true if one of these occurs. This limits the amount of warning messages to one (so that users can deal with one problem at a time), and enables the program to 'bail out' of the evaluation process mid-way, which acts as a time-saving feature.

'Example' is used to identify a world which can act as an example (or counter-example) for a claim. Once the user has been told whether the claim is true or false, they are offered the chance to view this world in order to illustrate why that truth value was given.

GridWorld: methods

Apart from the standard accessor and mutator messages, the GridWorld class methods are mainly concerned with keeping track of the pieces placed on the grid. This process requires methods for placing pieces, checking routines which ensure that pieces are not placed on top of each other or off the edge of the board, and methods for moving pieces around.

Additionally, GridWorld has a generateWorlds method. This handles the process of extending the Modal diagram by moving pieces around the grid, creating a new world for each resulting situation and updating the Universe model accordingly. Related methods deal with the naming of worlds.

The class GridWorld also contains the method 'evaluate: aClaim'. This takes a claim, splits it up, and acts as follows:

- If the first letter is a Modal operator, this is broken off, and a revised claim is sent on to all the worlds this one is connected to. These carry on the procedure until a truth value is returned. If the world is not connected to any others, an error message is given and the process stops.
- If this Modal operator is "necessity", the method returns 'true' if all the connected worlds return 'true', and false otherwise. If the Modal operator is "possibility", it returns 'true' if at least one connected world returns 'true'.
- If the first letter is *not* a Modal operator, the claim is evaluated for this world, rather than being passed on. If any of the objects that are being compared are not present in the world, an error message is given and the process stops. Otherwise, the value of the claim is returned.

A separate method is provided to deal with the problem of deciding which other worlds are accessible at any given time. The set of accessible worlds differs according to which Modal system is currently in use. In system T, this process simply returns the list of immediately connected worlds.

System S4 is transitive, however. In S4, building up a list of accessible worlds involves inspecting the list of connections for each connected world in turn, and adding any which are not already listed in the set. This process is repeated until no new worlds are added, at which point

the expanded set is returned. The effect of this is to compile a list of all the worlds further 'down' the tree from the starting position.

S5 is both transitive and symmetric. In this case, not only are the connections of each connected world examined, but the connections of every world held in the universe are inspected to see if they include elements from the set of worlds being interrogated. For example, if the set contains world 5, and world 3's list of connections includes world 5, then world 3 will be added to the set. Once no new worlds are added, the expanded set is returned. The effect of this is to gather all the worlds that are connected to the starting world in any way.

Universe

The Universe class introduces seven new variables: `worlds`, `worldToPlace`, `worldToDelete`, `worldToEvaluate`, `piece1`, `piece2`, and `piece3`. '`worlds`' is a dictionary, similar to '`pieces`' in the GridWorld class. This stores a list of grid worlds, together with co-ordinates that are used to position the world in the interface. '`worldToPlace`' and '`worldToDelete`' are Boolean variables which act as flags, letting the model know if the user wanted to add or delete a world.

'`worldToEvaluate`' works a little differently from the previous two variables. Modal claims cannot be made without specifying which world the claim is to be made for. This variable holds a record of the world the user has chosen.

The last three variables (`piece1`, `piece2`, and `piece3`) keep track of the pieces that the user wants to compare when evaluating a claim.

Universe's methods are primarily concerned with tracking and manipulating the worlds it contains. In addition, Universe keeps track of the pieces being compared in the Modal claims entered by the user.

(ii) Implementation of the Views

The superclass of GridWorldView and UniverseView was refined by re-implementing certain key methods. Neither required any new variables to do this.

Both views identify a dedicated Controller mechanism (GridController and UniverseController, described below), in preference to the generic one. Additionally, each has a unique '`displayOn:`' method. That of GridWorldView enables it to draw a grid, place pieces in the correct squares, and write their names underneath. That of UniverseView draws a rectangle for each world, writes the world's name inside, and draws a line to each of the worlds it is connected to.

The information for these operations is gathered by a variable inherited from the class View. When instances of the views are created, a variable called '`model`' is given the value of the underlying model, which can then be interrogated.

Figures 7 and 8 illustrate the two resulting views, complete with worlds and pieces.

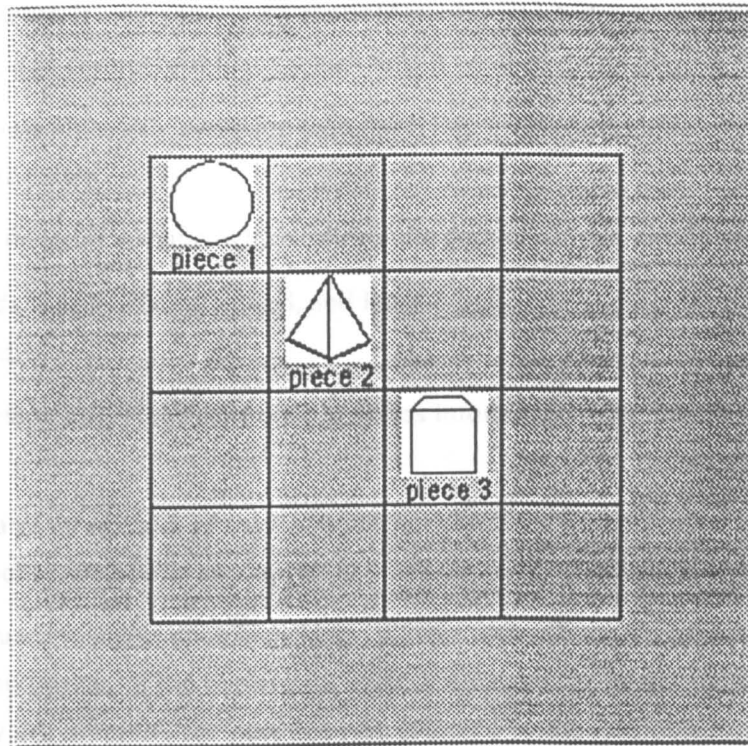


Figure 4.7: An example of the final GridView.

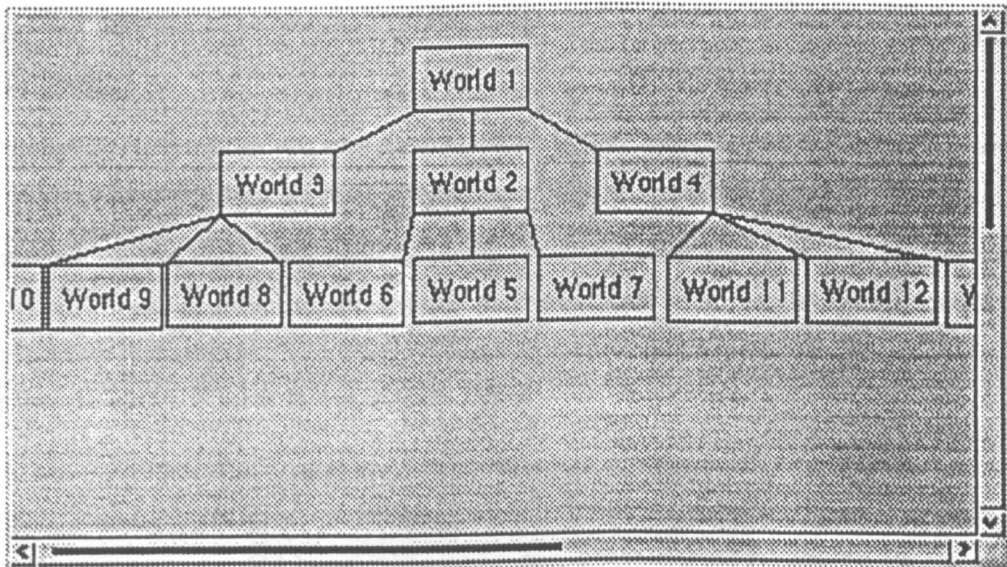


Figure 4.8: An example of the final UniverseView.

(iii) Implementation of the Controllers

MoLE's controller classes fell into two distinct categories: dedicated 'Controller' sub-classes to manage the use of the mouse, and the more general 'ApplicationModel' sub-classes, which were used to control the way the view accessed and responded to changes in the underlying model.

Part 1: 'Controller' sub-classes

GridController

As with the View subclasses, the GridController and the UniverseController work in very similar ways. Both describe the way the mouse is to act when the pointer is moved over a view; consequently, the controllers focus on three methods: `controlInitialize`, which tells the mouse

how to behave when it enters the view, `controlActivity`, which describes how it should behave inside the view, and `controlTerminate`, which tells it what to do on leaving the view.

`GridController` introduces six variables: `icon`, `carrying`, `gridWorldInterface`, `dispatcher`, `lastPosition`, and `centre`.

'`icon`' holds the image of a shape, which follows the pointer around as it is moved.

'`carrying`' records the piece being moved, and lets the controller know if there is a new piece to place on the grid.

'`gridWorldInterface`' allows the controller to access information from the interface model (see below).

'`dispatcher`' creates an instance of the `UIDispatcher` object for the controller; in essence, this allows the controller to respond appropriately when the user double-clicks the mouse button.

'`lastPosition`' is used to detect double-clicking. When a mouse button is pressed, its location is compared to `lastPosition`. If the two are the same, the dispatcher carries out its double click activity. If they are different, the new position is stored in the `lastPosition` variable.

Finally, '`centre`' stores the co-ordinates in the View of the centre of the grid. This is calculated by scaling the mid-point of the `GridWorld` class' grid size by a factor of 22 (the width of each square in pixels).

Additionally, `GridController` has one class variable: `Blank`. This is used to store an empty image. When a mouse button is pressed in an empty square, this blank image is returned.

As mentioned, the two controllers focus on methods that are concerned with interpreting mouse actions. `GridController` initialises the pointer as it enters the view, changing it to look like a hand to indicate that pieces can be dragged-and-dropped in this window (Figure 4.9). On leaving the view, this is reset to the default arrow.

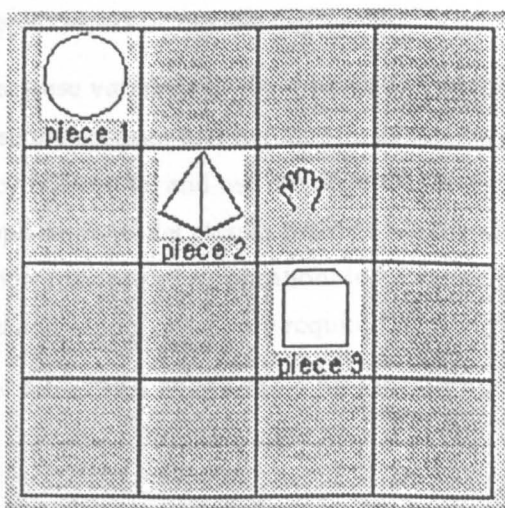


Figure 4.9: The hand icon

The `controlActivity` method deals with mouse button activities. It detects double-clicking, and manages the methods which tell whether a piece is picked up when a button is pressed. If a piece is picked up, this also controls the method that places it back on the grid when the button

is released. Whether a piece is picked up, and where dropped pieces are placed, is determined by methods which read the pointer's position in the view, establish the position relative to the centre of the grid, and then re-scale from pixels to squares in order to calculate which square the pointer is in.

UniverseController

UniverseController introduces seven new variables: world, universeInterface, dispatcher, lastPosition, carrying, icon, and offset.

'universeInterface' operates in the same way that 'gridWorldInterface' works for GridController; 'carrying', 'dispatcher' and 'lastPosition' also fulfill the same roles. 'world' keeps track of the world selected for drag-and-drop or double-clicking. 'icon' creates a rectangle, the same size as the selected world, which follows the cursor point around during drag-and-drop operations.

'offset' is also to do with drag-and-drop displays. Normally, the top left hand corner of the icon would follow the pointer. 'offset' records the difference between the top left hand corner and the point at which the mouse button is depressed, allowing for a more natural drag-and-drop appearance.

UniverseController's methods are similar to that of GridController, supporting mouse activities.

Part 2: 'ApplicationModel' sub-classes

VisualWorks encourages software reuse. Because of this, applications adopting the Model-View-Controller paradigm will typically include objects whose purpose is to link the interface to the underlying model, in addition to those dedicated to supporting the use of the mouse or keyboard. These objects are created as sub-classes of ApplicationModel, a further refinement of the Model class. They allow different interfaces to be designed for use with the one model, or alternatively, different models to be attached to one interface, by providing an object that can be easily altered to 'plug' different views or models together.

ApplicationModel sub-classes use variables of the type ValueHolder to keep track of variables stored in other objects. These use accessor and mutator methods ("value" and "value:") to alter the value of the variable they hold on to, and use the dependency mechanism to update displays. Additionally, ApplicationModel classes provide functionality for buttons on the interface. Finally, when a new instance of ApplicationModel is made, it creates instances of whatever Views, Controllers and Model objects are required, and then connects them.

Two such ApplicationModel sub-classes were created for MoLE: GridWorldInterface and UniverseInterface.

GridWorldInterface

GridWorldInterface introduces seven new variables: gridWorld, gridWorldView, gridNameHolder, editNameHolder, editPiece, editShapeHolder, and connectionsList.

'gridWorld' and 'gridWorldView' allow the object to identify the components it connects.

Variables with the suffix, '-Holder', are of the type ValueHolder described above. These keep track of variables, updating displays and responding to user inputs. Specifically, these allow users to edit the name of each grid, and to use a dialogue window to specify the name and shape of the pieces on the grid.

'editPiece' keeps track of the piece selected for editing. This is then accessed through the editing dialogue window, which is launched by double-clicking on a piece.

'connectionsList' acts in a similar way to a View, providing a display which lists the worlds to which this grid is connected. Although updating the list still requires the use of the dependency mechanism, this type of object has the benefit of allowing users to select list elements which can then be accessed by, for example, button actions.

Due to the specific job these classes perform, most of GridWorldInterface's methods are to do with interface support. These methods fall loosely into three categories: updating text and list windows, building menus, and supporting buttons. These methods allow users to copy grids from one world to another, wipe grids clean, add, edit or delete pieces, add or remove connections, and open highlighted connections from the list. The remainder of the methods in this class pass information or messages from the interface to the model.

Figure 4.10 illustrates the window supported by the class GridWorldInterface.

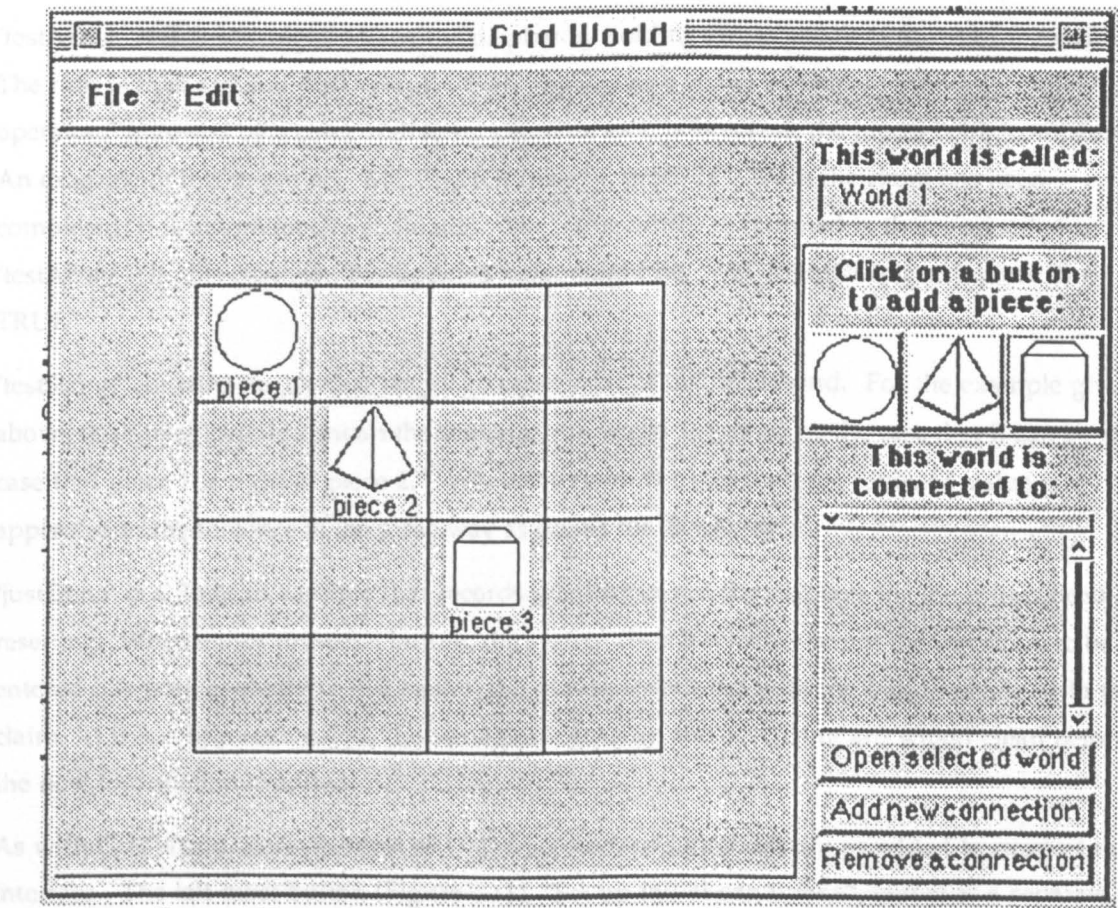


Figure 4.10: An example of a window supported by GridWorldInterface

UniverseInterface

UniverseInterface introduces 13 new variables: `universe`, `universeView`, `gridX`, `gridY`, `onlyCirclesMove`, `system`, `chosenWorld`, `object1`, `object2`, `object3`, `testArray`, `testString`, and `justDone`.

'`universe`' and '`universeView`' are used to connect the controller to the underlying model and the associated view, so that information can be passed from one to the other.

'`gridX`', '`gridY`', '`onlyCirclesMove`', and '`system`', are variables which control the way that grid worlds will be created and that new grid are worlds derived when pieces are moved. They are accessed through the "Set up initial conditions" dialogue window. The first two variables allow the user to change the size of grid worlds by specifying new dimensions; these are restricted to being between 1 and 6 squares in size, as described in section 4.3. '`onlyCirclesMove`' supports a check box, and stops any piece other than spheres from moving. '`system`' is used to control a set of three radio buttons, one each for the Modal systems T, S4 and S5, by keeping track of which Modal system the user has selected. It is also accessed when evaluating claims, in order to calculate the set of accessible worlds.

The remaining variables support the process of evaluating Modal claims. '`chosenWorld`' is used to identify which grid world the claim is being made about. '`object1`', '`object2`', and '`object3`' identify the pieces being compared by the claim.

'`testArray`' and '`testString`' support the display by holding two equivalent forms of the claim. The first is in shorthand (`testArray`), which appears as a string of letters (denoting Modal operators) followed by a comparative proposition, and, in brackets, the pieces being compared. An example of this might be, "MC RightOf (piece 1, piece 2)". Once the claim has been completed, it is automatically evaluated. The value which is returned is then appended to '`testArray`'. In this example, the updated value might be, "MC RightOf (piece 1, piece 2) - TRUE".

'`testString`' contains the natural language equivalent of this shorthand. For the example given above, '`testString`' would contain the sentence, "In world 1, it must be the case that it can be the case that piece 1 is right of piece 2." When the claim is evaluated, the returned value is appended to the sentence in the same way as occurs for '`testArray`'.

'`justDone`' is a Boolean variable that records whether or not the displays for the claims need resetting. When a user presses a button, the system checks whether or not the last claim to be entered has been evaluated. If it hasn't, the phrase associated with the button is added to the claim. If it has been evaluated, the variables `testArray` and `testString` are wiped clean before the new information is added.

As with `GridWorldInterface`, most of `UniverseInterface`'s methods are provided to support the interface. The left-most button (Figure 4.11), "Set up initial conditions", launches a separate dialogue window that enables users to alter the grid size by editing numbers in text windows, and specify whether or not spheres are the only pieces allowed to move. The user can also switch

between the three Modal systems by using a set of radio buttons. The second button on the UniverseInterface resets the universe model by wiping all the worlds from the 'worlds' dictionary. The third and fourth support the creation and deletion of worlds.

In addition to the buttons dealing with grid worlds, eight further buttons allow sentences to be manipulated. The panel of buttons in the bottom-right hand corner of the interface (Figure 4.11) allows sentences to be built up or canceled. A dialogue window is then launched which asks users to select the pieces they want to compare. The user can choose which world to evaluate the sentences in by selecting one from a pop-up menu. The user is asked to confirm their choices before the object then sends the completed sentence to the selected GridWorld to be evaluated.

The method, 'offerExample: anExample', also relates to the process of evaluation. When evaluations are carried out, cases which prove a 'can' claim, or which disprove a 'must' claim, are offered as examples to users. When a claim has been evaluated, a dialogue window asks users if they want to see the relevant example. If they accept, a window is opened on the grid world which proved or disproved the claim in order to illustrate why it succeeded or failed.

Figure 4.11 illustrates the window supported by the class UniverseInterface.

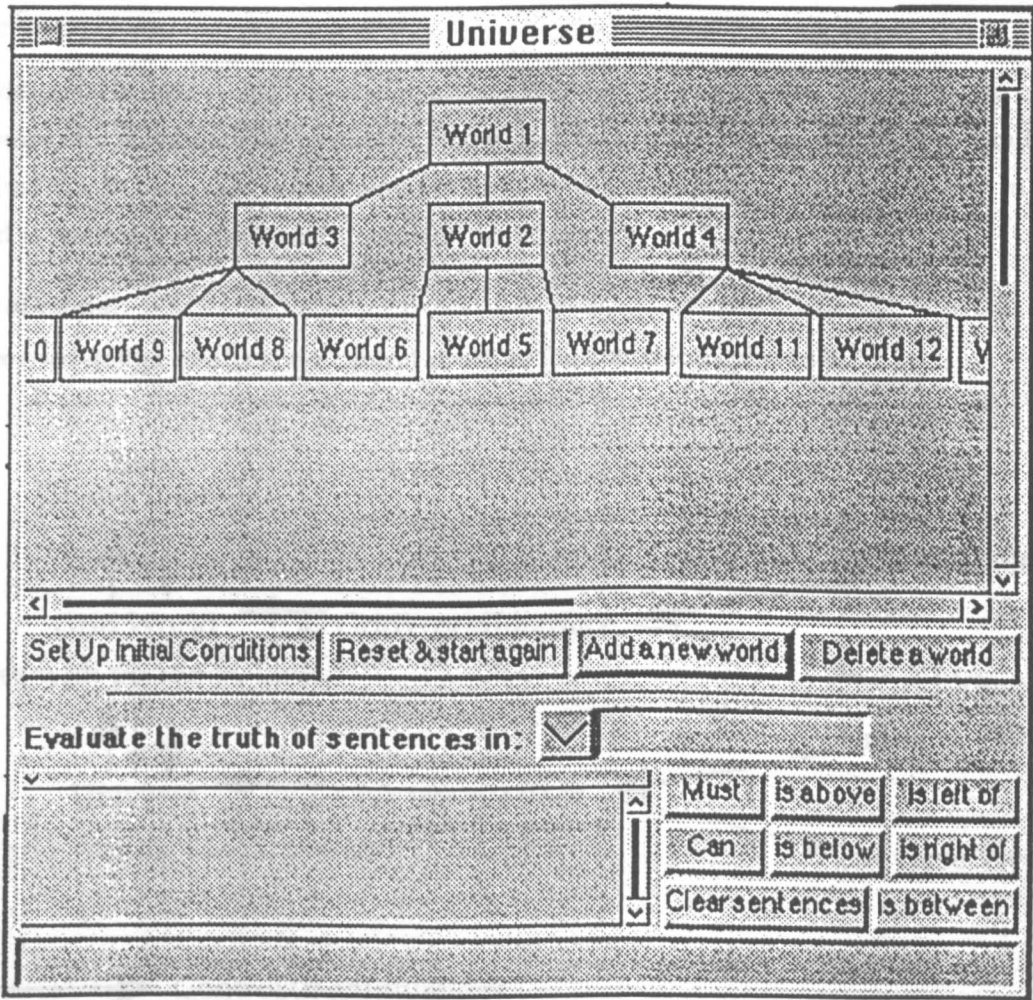


Figure 4.11: An example of a window supported by UniverseInterface

Figure 4.12: Flow of control diagram for the revised design of MULE

Design Refinements

As mentioned in the section on the GridWorld model, two elements of the original design were altered during implementation. The first change involved replacing the array, which was intended to keep track of pieces' positions, with a dictionary. The other was to introduce a variable that allowed grid worlds to access the universe they were a part of.

In addition to these alterations, one other feature was changed, with important consequences. The original design involved each grid world containing a set of pieces which would then be placed on a grid. During implementation, it was decided that a simpler and more efficient design would be to have all grid worlds access the same set of pieces. Since the number of grid worlds increases exponentially when pieces are allowed to move, this simplification provides a considerable reduction in the number of objects required.

The revised design also permits one Modal topic to be avoided: the problem of reference. Having all grid worlds hold the same set of pieces makes it very difficult to create the type of situation where pieces with the same name have different properties in each world, or else fail to appear. These are examples of the problem of reference in operation. With the original design, this problem would have been common; even minor typographical errors would have given rise to reference problems.

An additional benefit of this refinement can be seen for situations where a user decides to change the properties of a piece *after* creating a Modal diagram. In the original design, they would be required to edit that piece in each world in turn. With the revised design, an alteration in one world is automatically reflected in each of the others.

Flow of control

The revised design results in the control structure illustrated in Figure 4.12.

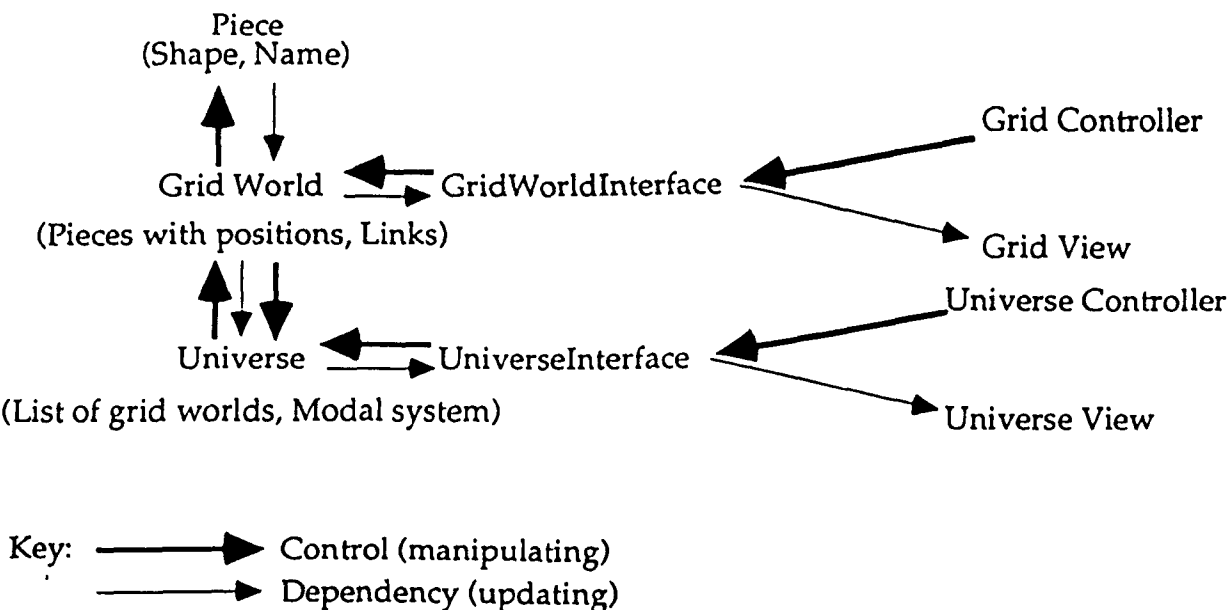


Figure 4.12: Flow of control diagram for the revised design of MoLE

Three features distinguish this diagram from that in Figure 4.6. Firstly, a link from Grid World to Universe has been added, allowing the Grid World class to send messages to the Universe, in addition to sending updates via the dependency mechanism. Additionally, the ApplicationModel classes GridWorldInterface and UniverseInterface have been adopted as intermediaries between the model and the views. They also interact closely with the classes GridController and UniverseController, which manage mouse operations.

4.6 Conclusions

This chapter has outlined the design and implementation of a software tool for learners of Modal logic.

A learning environment was chosen as an appropriate visualisation and manipulation tool to support students learning Modal logic. This design supported four key features: interactivity, constraint of errors, the ability to offer contextualised examples, and the automation of rote tasks. Section 2.3 indicates that the third and fourth of these are design features common in existing software for students learning logic.

The learning environment representation used in the first study was adopted as the basis of the software tool. In order to adapt it to this purpose, however, it was necessary to revise the representation in several important respects. The grid worlds were separated from the diagrams describing the relationship between worlds, the types of pieces were simplified, and the size of grids was restricted. This satisfied both pragmatic concerns of screen constraints and pedagogic issues of representation use.

A discussion of design aims demonstrated that the Model-View-Controller mechanism formed a suitable basis for the implementation of MoLE. However, the aims necessitated a revision to the mechanism which allowed it to support a hierarchy of models and views (Figure 4.6).

The process of implementing MoLE led to several design revisions, including:

- further modification of the Model-View-Controller mechanism to allow the Grid World models to access the Universe model.
- use of ApplicationModel sub-classes as a model-view link.
- a shared 'set' of pieces, rather than a separate set for each GridWorld, which allowed potential problems with the Modal topic of reference to be avoided.

An overview of MoLE's intended use is provided in the next chapter, where a typical problem is worked through, step by step. A formative evaluation of MoLE will also be described, which will assess the extent to which the software meets its design aims.

Chapter 5

An Overview and Formative Evaluation of MoLE

5.1 Introduction

This chapter provides an overview of the functionality of the software tool MoLE. It does this by providing a step-by-step illustration of the way in which MoLE can be used to answer a typical Modal logic question.

A formative evaluation of the tool is then presented, and the revisions introduced as a result of the study are outlined. The study involved five subjects working through one Modal logic question, and then exploring the limits of the software tool in an effort to test how robust MoLE was.

The chapter concludes with a summary of how the revised tool differs from the design described in chapter four.

5.2 An overview of the use of MoLE

This section describes how MoLE can be used to explore problems in Modal logic. Working through a typical Modal logic question illustrates how MoLE can be used, and demonstrate the features MoLE offers.

An example is provided of the kind of problem MoLE can be used to solve, and the steps required to complete it are outlined. These steps can be grouped into three stages: establishing the initial conditions, constructing the diagram, and evaluating claims.

The problem being investigated

The problem used for this example involves investigating for which systems the claim, “it must be the case that piece 1 is above piece 3”, is true for a 4x4 grid world. In this world, spheres are

allowed to move up to one square up, down, left or right, subject to the usual restrictions. The world contains the following pieces:

- A sphere called piece 1 in the top left hand corner of the grid.
- A tetrahedron called piece 2, one square right and one down from piece 1.
- A cube called piece 3, two squares right and two down from piece 1.

A total of two moves are allowed.

Establishing the initial conditions

Initially, the user is presented with a new Universe window. Their first task is to use the “set up initial conditions” dialogue to ensure that the grids are of the size required by the question. Figures 1 and 2 illustrate this.

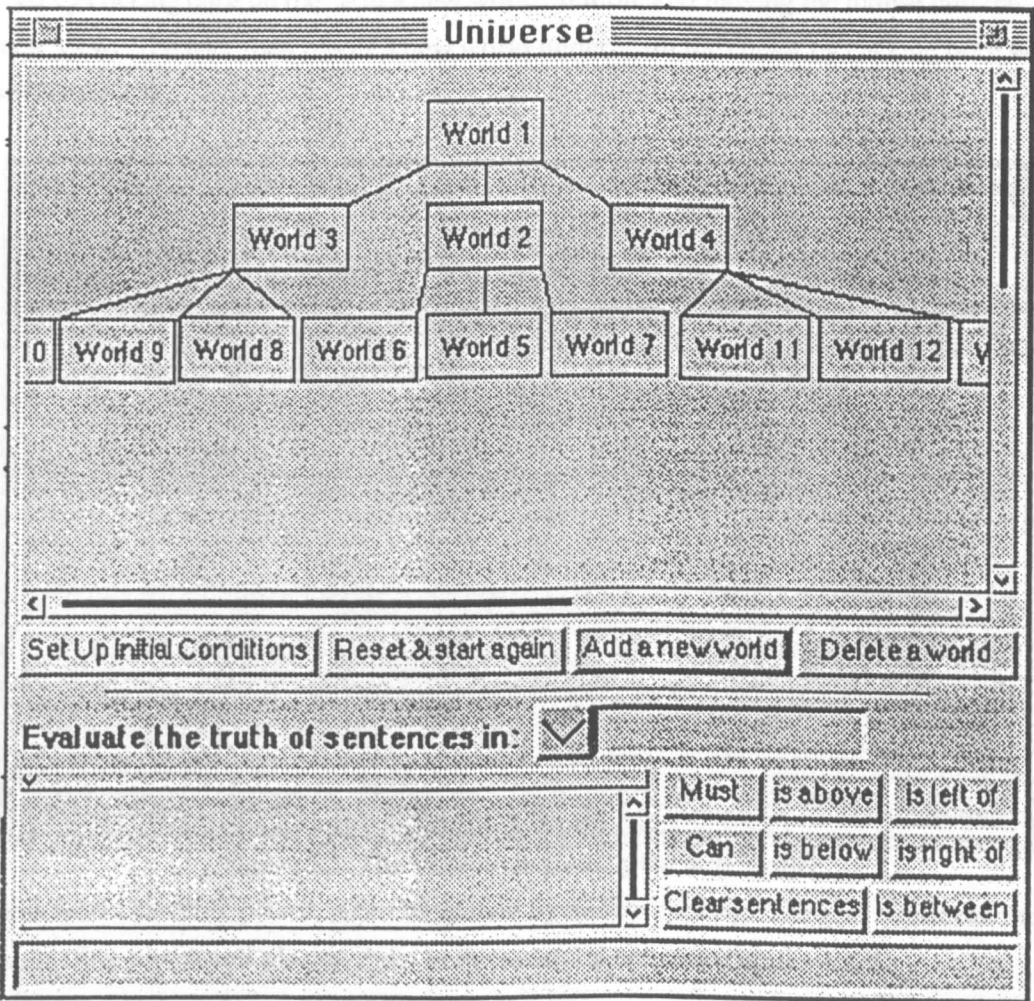


Figure 5.1: The initial Universe window

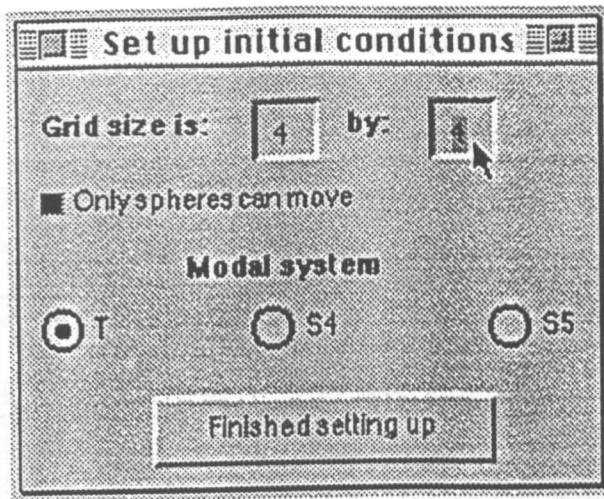


Figure 5.2: Using the “set up initial conditions” dialogue to set the grid size

Once this has been done, the first grid world can be placed in the Universe view. This is achieved by clicking once on the “add a new world” button, and moving the mouse into the Universe view. The user then places a new world underneath the cursor by clicking once with the mouse (Figure 5.3).

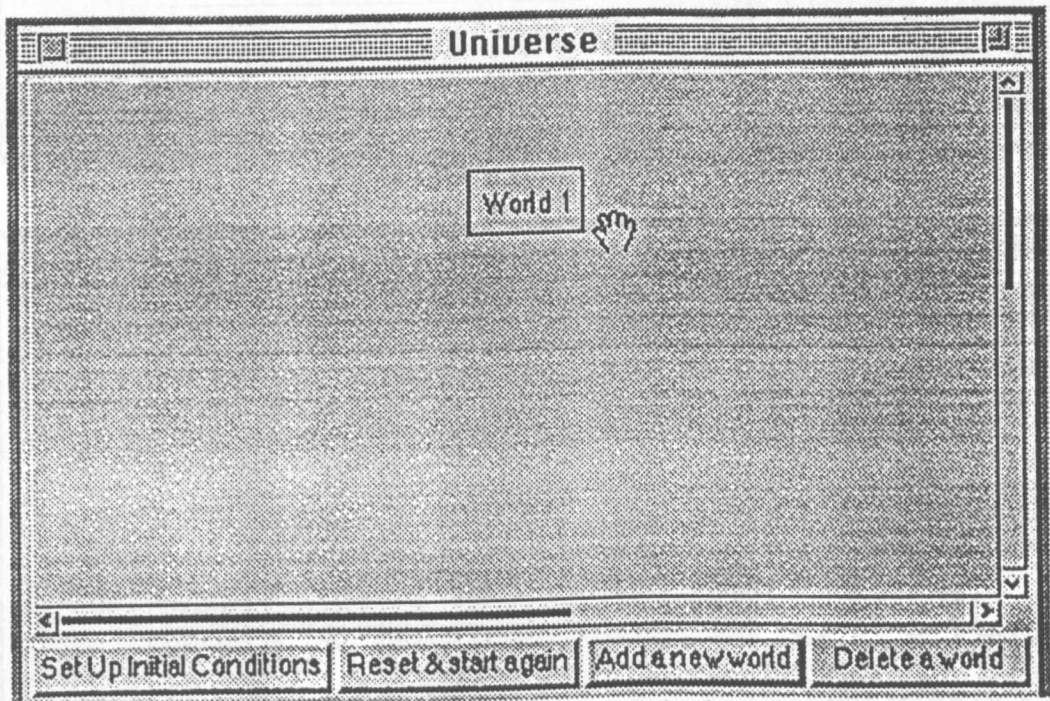


Figure 5.3: The view after placing the first world

Once this world has been placed, it can be opened by double clicking. The grid world can then be edited, creating pieces and placing them in the specified squares (Figures 4 & 5).

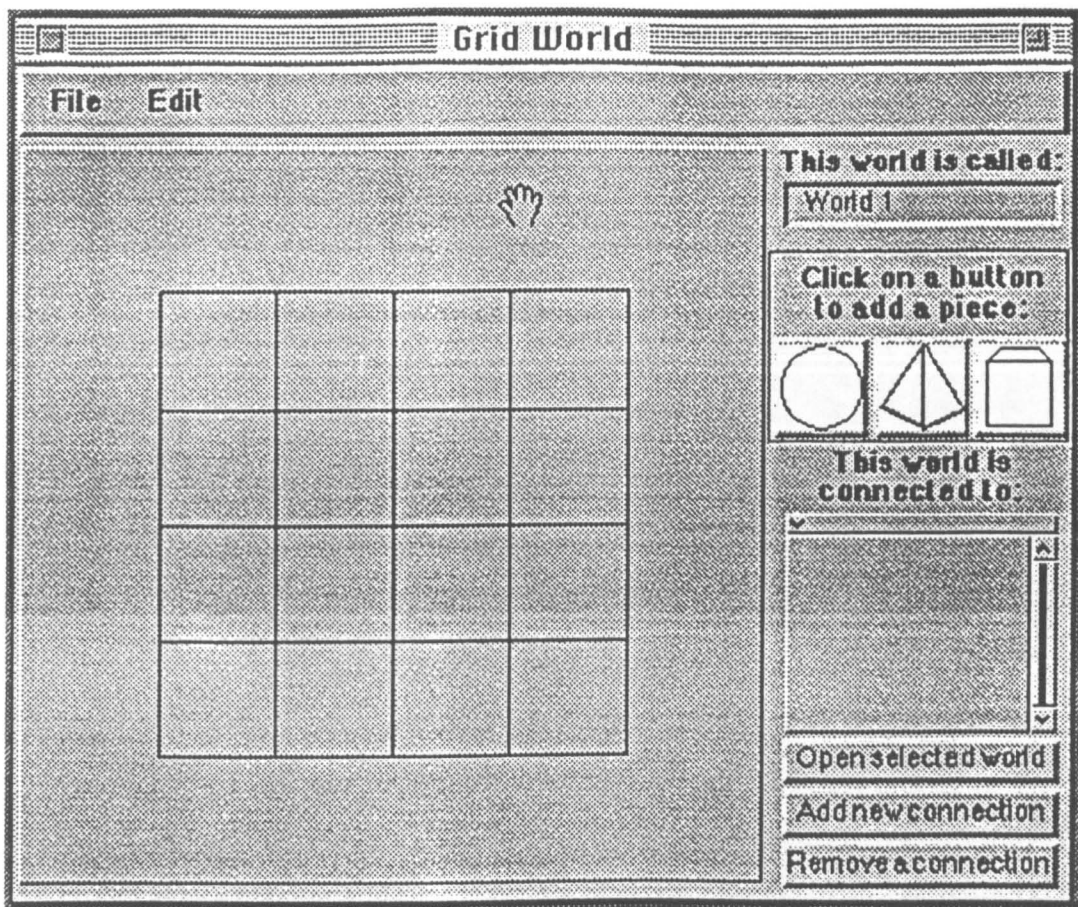


Figure 5.4: The grid world window

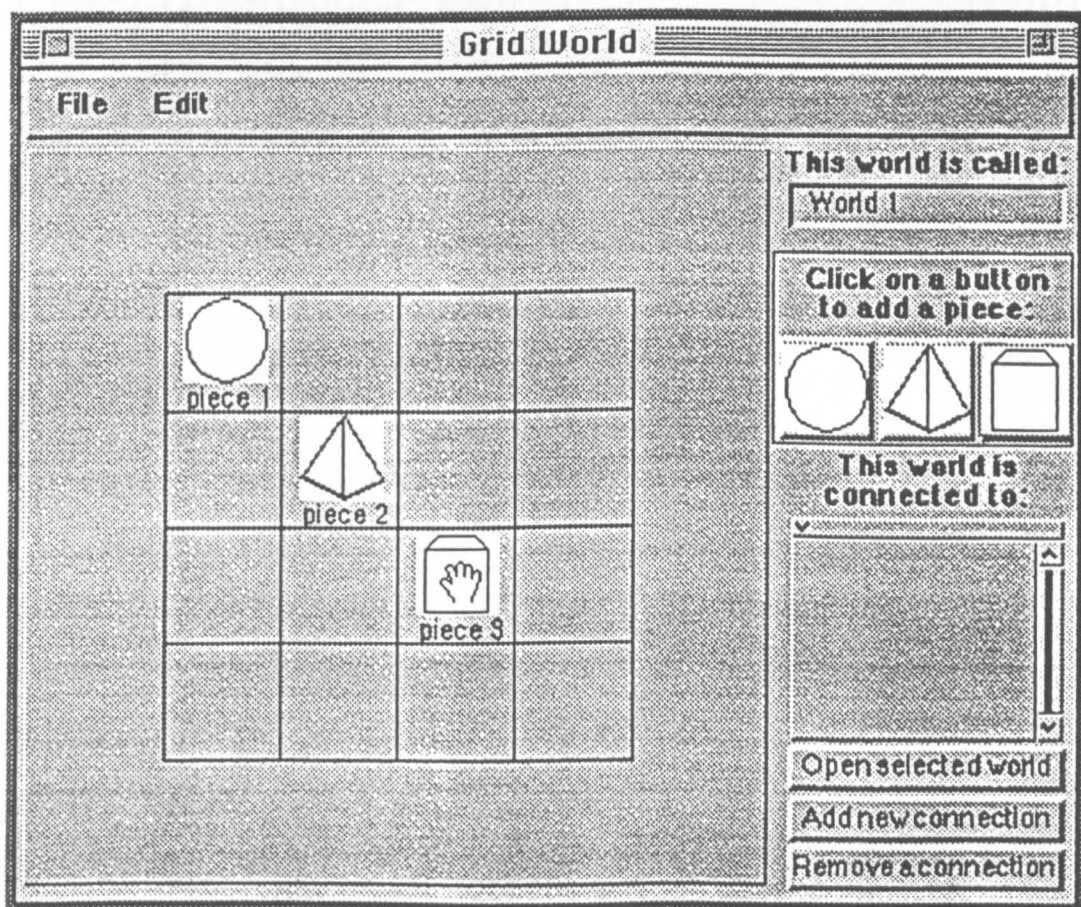


Figure 5.5: The grid world window, with pieces placed

At this point, all the initial conditions have been set up as required by the question, and the diagram, which will form the basis of the proofs, can be constructed.

Constructing the diagram

Constructing the Modal diagram is achieved by using the 'move pieces and create worlds' command from the grid world's 'edit' menu (Figure 5.6). This creates the first layer of a Modal diagram (Figure 5.7).

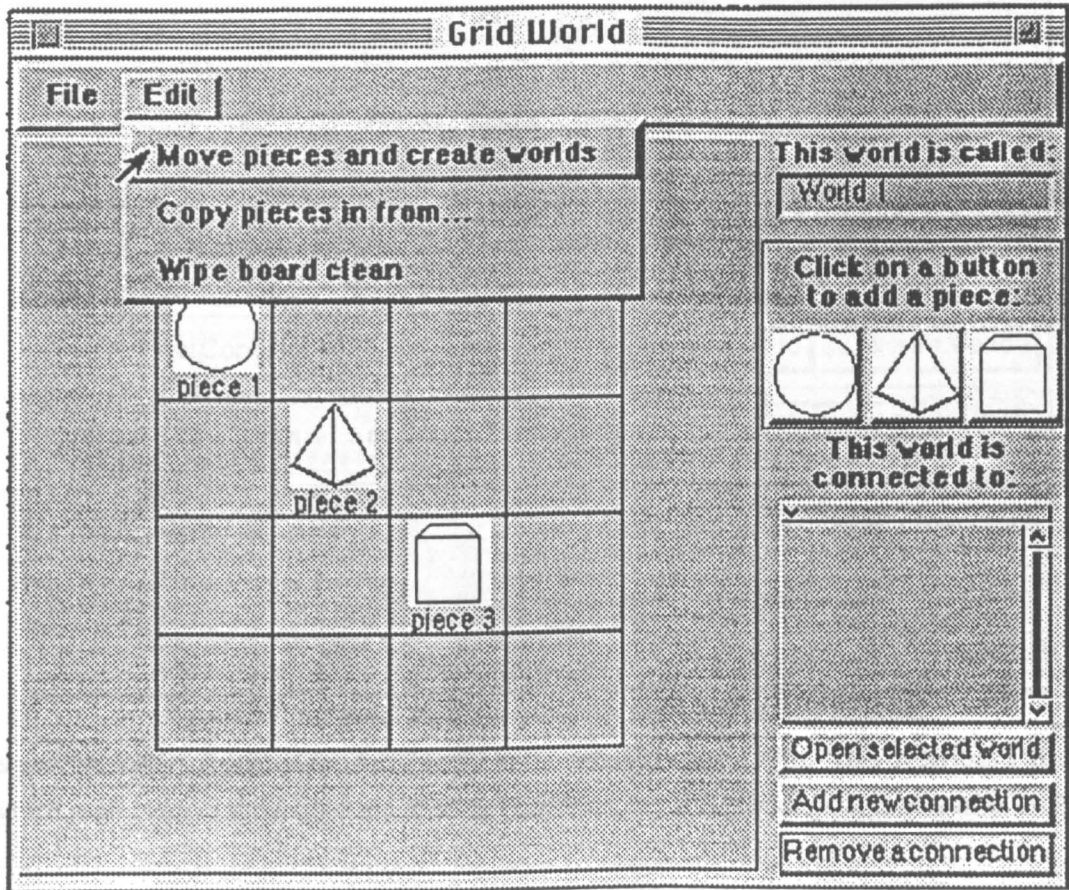


Figure 5.6: Using the edit menu to create a new Modal diagram

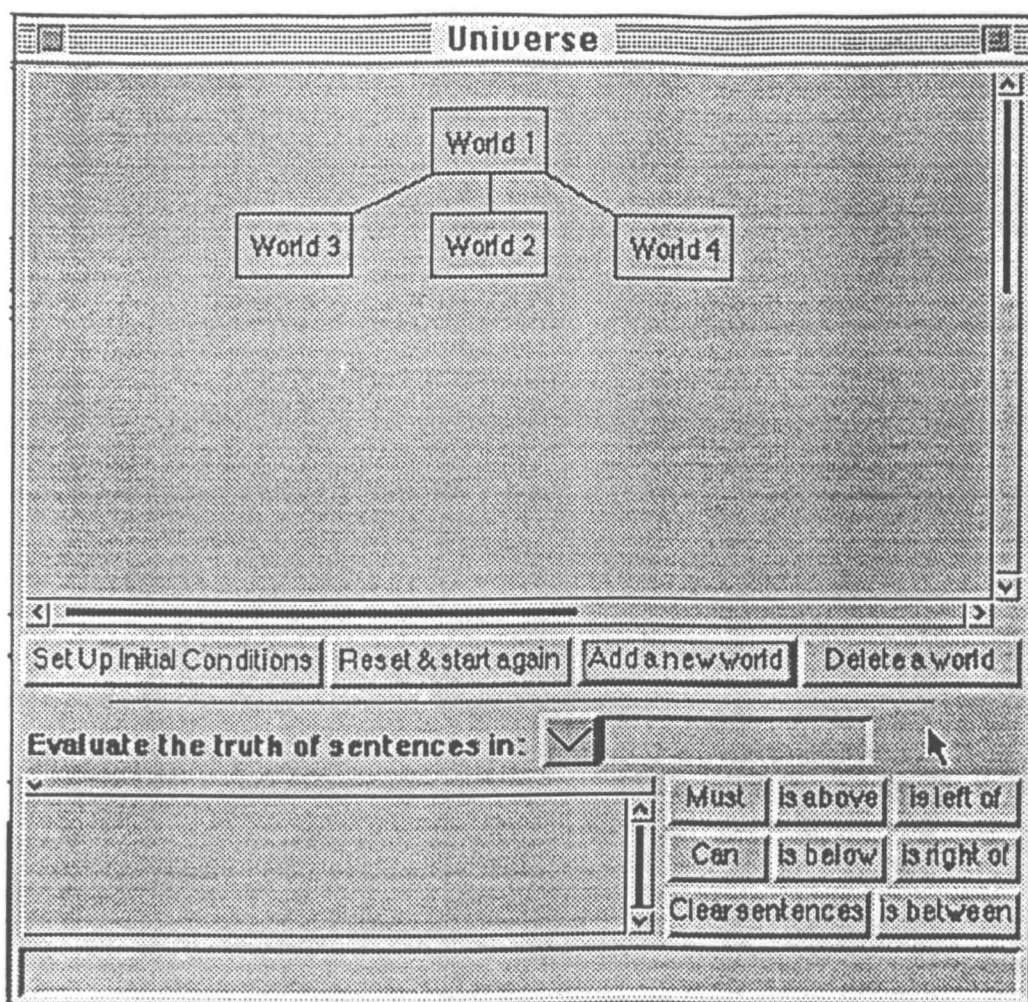


Figure 5.7: A Modal diagram in which pieces have moved once.

This process is then repeated for each world in the second layer in turn. This describes a situation in which each piece is allowed to move up to two times, resulting in the diagram shown in Figure 5.8.

To evaluate the truth of a sentence, the user must first select a world for which the claim is to be evaluated. This is then indicated in the text field to the right of the label, "Evaluate the truth of sentences in:" (Figure 5.9).

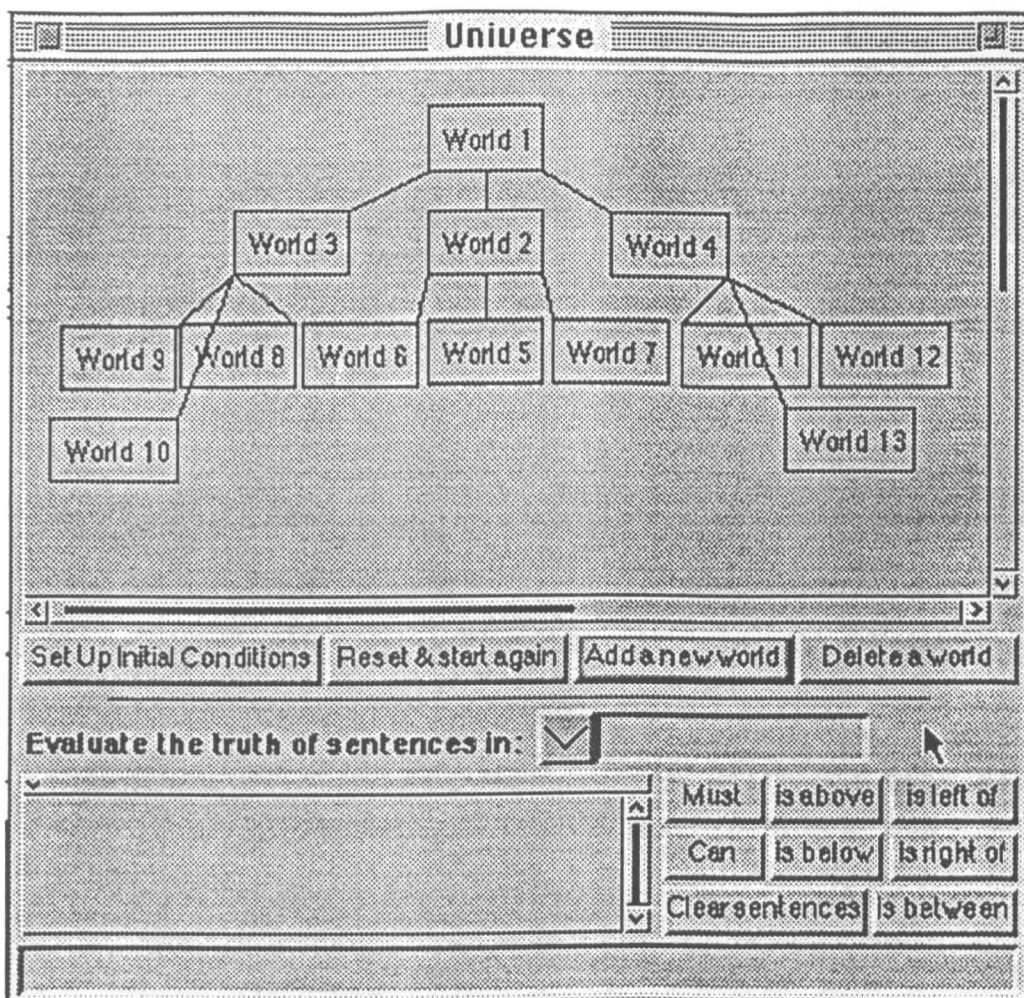


Figure 5.8: The resulting Modal diagram

Evaluating Claims

Once the diagram has been constructed, it can then be interrogated to see when the claim, "it must be the case that piece 1 is above piece 3", is true. To do this, the user first decides for which world the claim is to be evaluated. This is then indicated in the text field to the right of the label, "Evaluate the truth of sentences in:" (Figure 5.9).



Figure 5.9: Building the claim

As the button "is above" is pressed, a dialogue window is opened. This lets the user select which pieces are to be compared by using pop-up menus. These pop-up menus are in the same format as the one used to select the world in which the claim is evaluated (Figures 11 & 12).

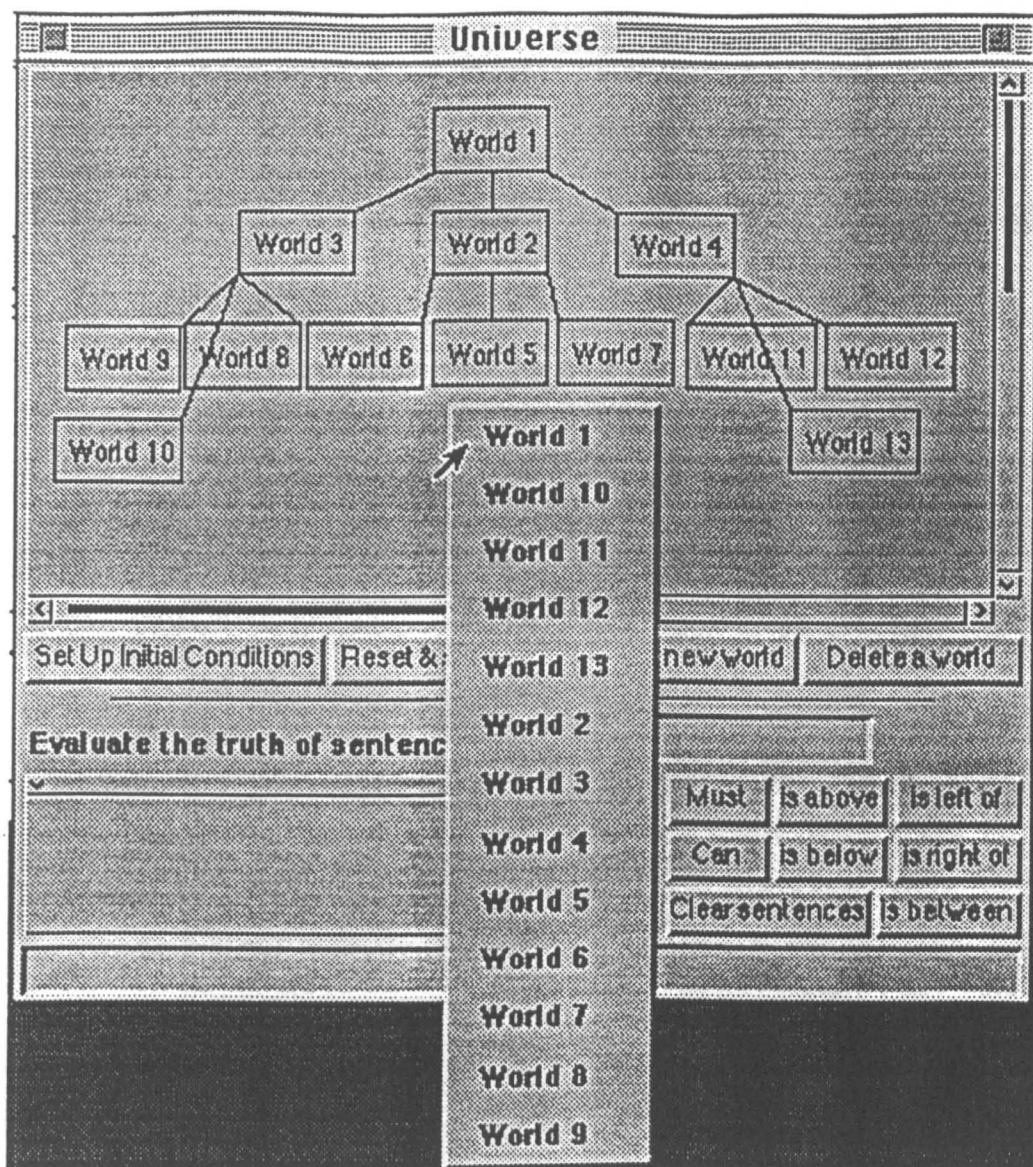


Figure 5.9: Selecting the world in which the claim is to be evaluated

After selecting a world, the sentence creation buttons can be used to build up the claim (Figure 5.10).

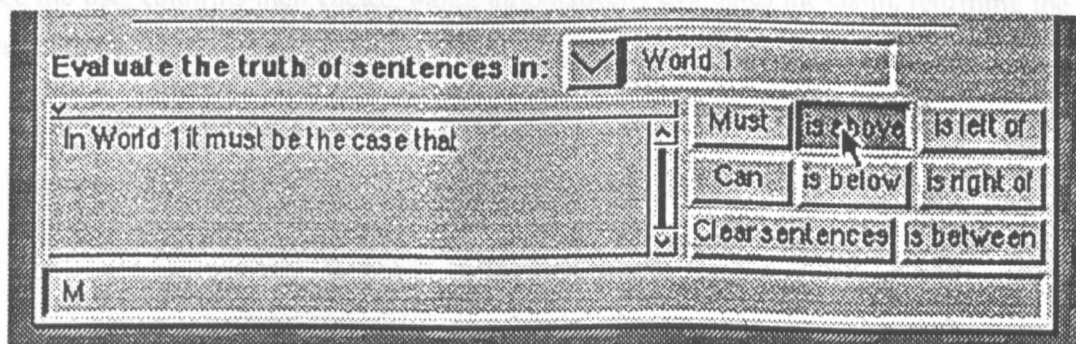


Figure 5.10: Building the claim

As the button "is above" is pressed, a dialogue window is opened. This lets the user select which pieces are to be compared by using pop-up menus. These menus are in the same format as the one used to select the world in which the claim is evaluated (Figures 11 & 12).

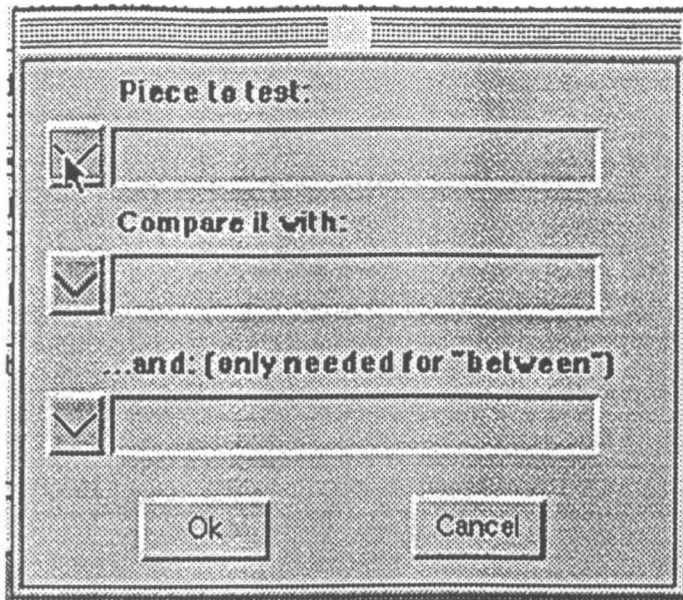


Figure 5.11: The dialogue window for selecting which pieces to test

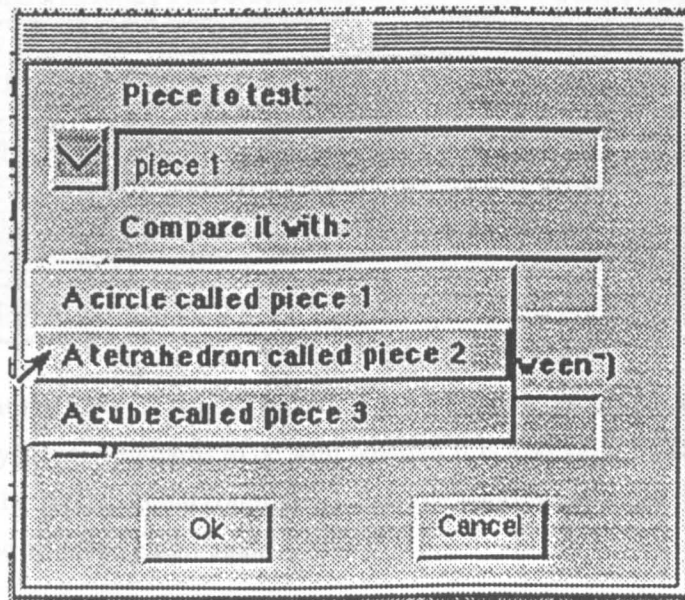


Figure 5.12: Entering the choice of pieces

Once the user confirms their choice, MoLE automatically evaluates the claim, returning the truth value (Figure 5.13).

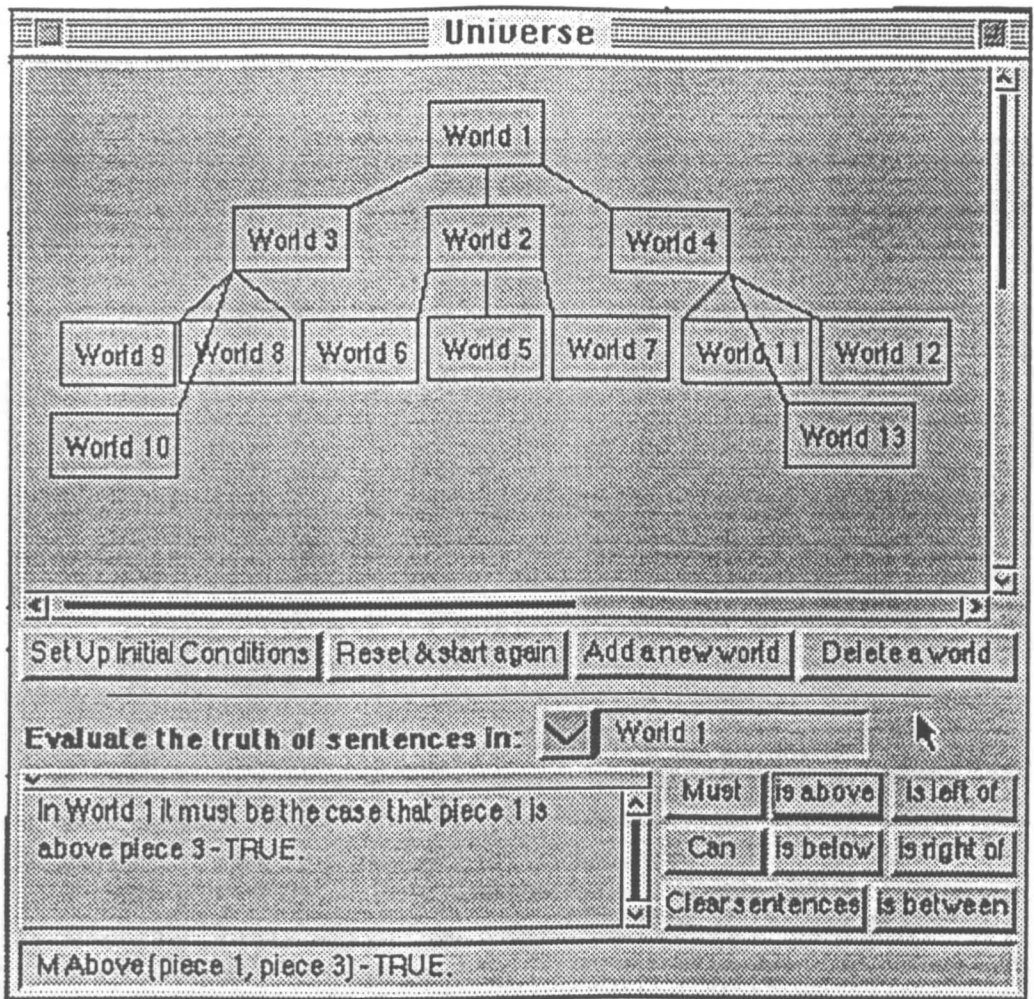


Figure 5.13: The truth value of the claim

This establishes that the claim is true in system T. Having now investigated the claim for the first of the Modal systems, it remains to be seen whether it is also valid in S4 and S5. To do this, the user returns to the “set up initial conditions” dialogue window, changes the current system to S4 (Figure 5.14), and tries the claim again.

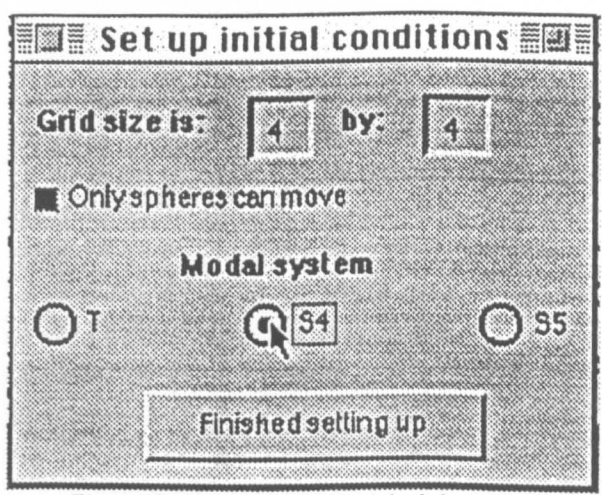


Figure 5.14: Altering the Modal system

Once the system has been altered, the claim can be re-entered, following the steps described above. This time, MoLE reports that the claim is false. Because this is a claim about necessity,

only one counter-example is required to disprove it. MoLE offers users the chance to inspect this counter-example (Figure 5.15).

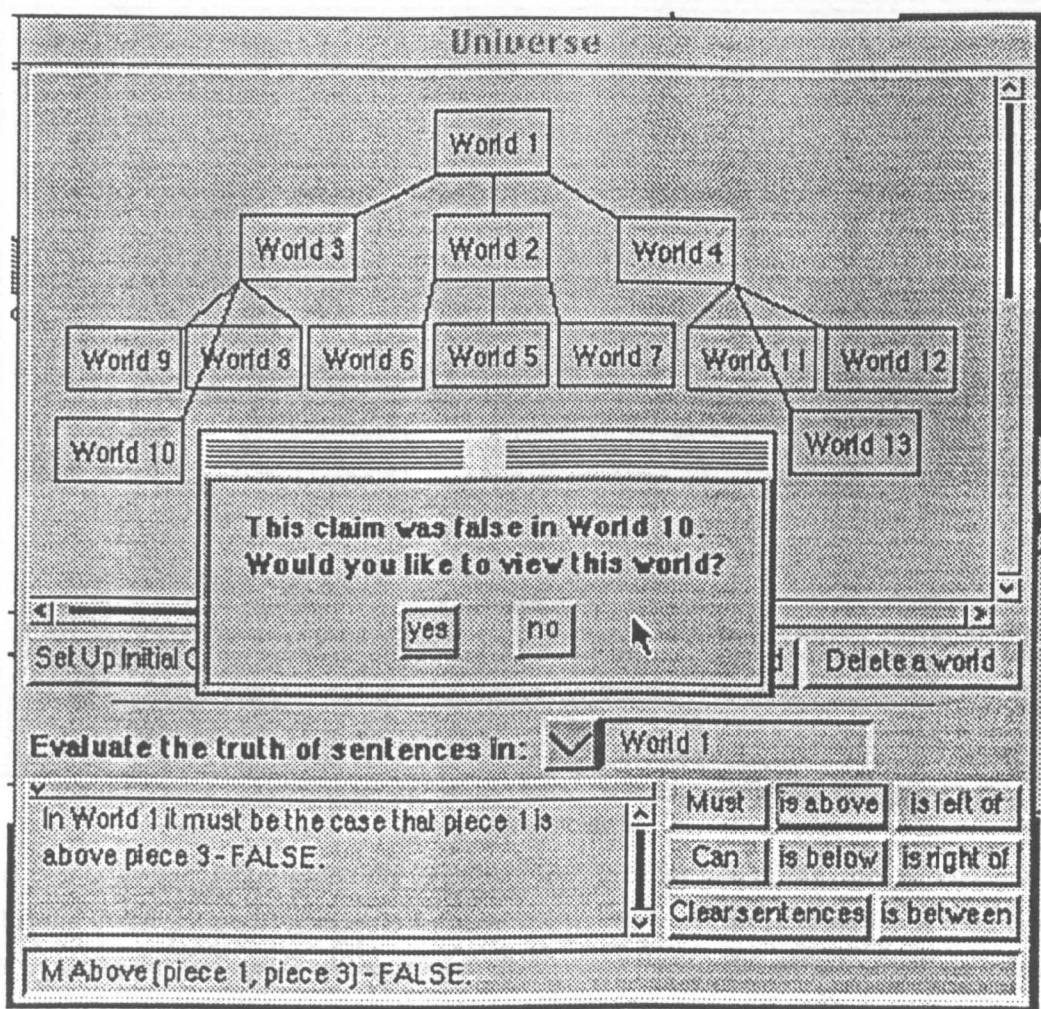


Figure 5.15: MoLE offers the users the chance to inspect a counter-example

If the user accepts the offer, a new window is opened on the relevant grid world (Figure 5.16).

ascertained that the claim is true for system T, but false for S1 and S5.

Summary

This section has shown how MoLE can be used to work through a typical question for Modal logic. This process involves setting up initial conditions, having MoLE extend the diagrams as far as is required, and then using it to assess claims.

5.3 A Formative Evaluation of MoLE

A formative evaluation of the software tool MoLE was carried out in order to refine the design and identify bugs in the program. In this section, the design and outcomes of the study are described.

Design

Volunteers took part in a formative evaluation of MoLE. None had any prior experience of Modal logic, but all had considerable experience of using computers. The evaluation took the

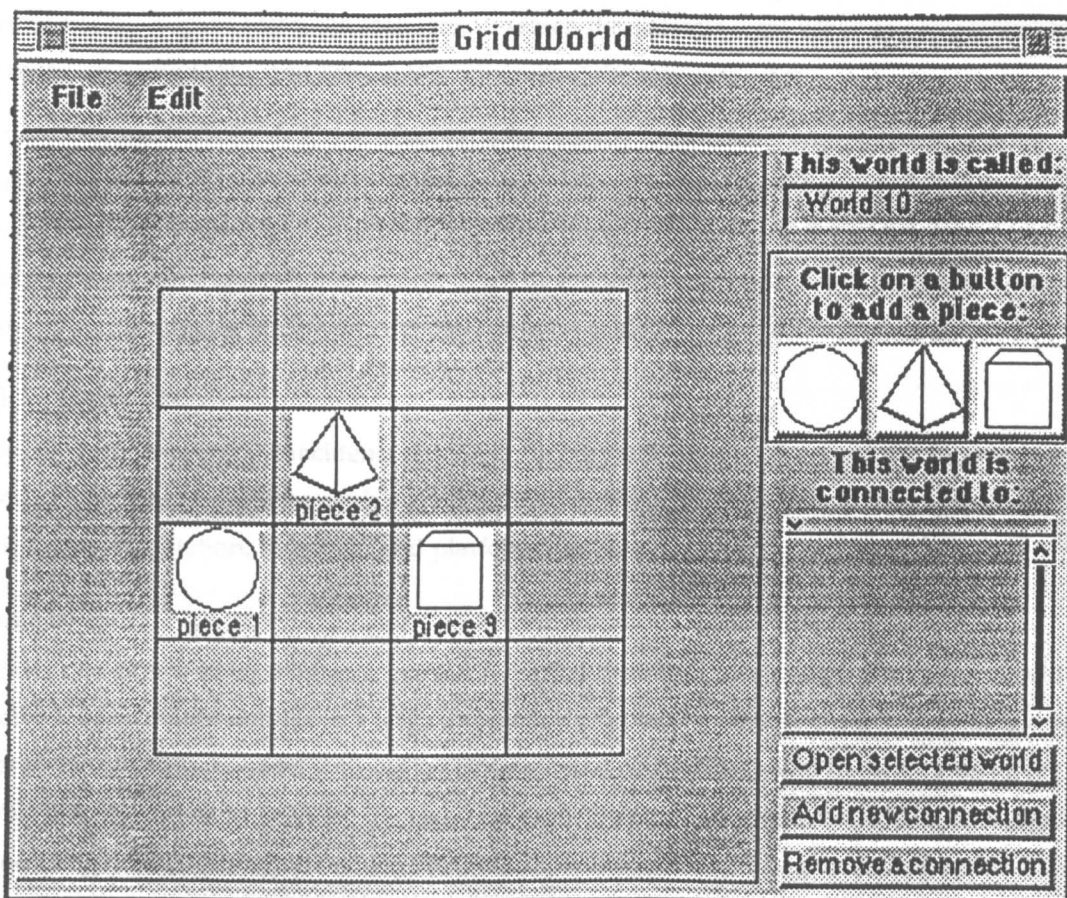


Figure 5.16: A counter-example to the claim, "it must be the case that piece 1 is above piece 3", in S4.

At this point, the user has successfully evaluated the claim for two of the three Modal systems, establishing that it is true in system T, but false in S4. The process of evaluation is then repeated for S5, by using the "set up initial conditions" dialogue to change the system, and then following the same procedure that was used for the previous two systems.

Once this has been done, the user will have successfully completed the exercise, having ascertained that the claim is true for system T, but false for S4 and S5.

Summary

This section has shown how MoLE can be used to work through a typical question for Modal logic. This process involves setting up initial conditions, having MoLE extend the diagrams as far as is required, and then using it to assess claims.

5.3 A Formative Evaluation of MoLE

A formative evaluation of the software tool MoLE was carried out in order to refine the design and identify bugs in the program. In this section, the design and outcomes of the study are described.

Design

3 volunteers took part in a formative evaluation of MoLE. None had any prior experience of Modal logic, but all had considerable experience of using computers. The evaluation took the

form of a session spent using the software, in which the subjects were asked to complete at least one question using MoLE and then to test how robust the software was.

Sessions were video recorded, and observational records were taken. (Appendix 7 contains observational logs for the session.) Subjects were encouraged to talk through difficulties they encountered and to comment on any aspects of the design which they felt could be improved upon. The comments and observational records were then used to compile a list of problems with the software and to suggest improvements that could be made.

5.4 Results

Subjects' comments and the incidents observed during the session were coded and used to identify several categories of problem, which are discussed in this section. These included software design, implementation, interface, and platform issues.

Design issues

The creation of one-layer diagrams proved remarkably simple, even when the blocks-world involved was complex and led to a large disjunction of possible worlds. However, adding a second layer to the diagram required subjects to open each and every new world in turn, re-using the "move pieces" command from the file menu for each in turn. This problem could be addressed by introducing further commands which allow multiple movements to be performed, rather than just supporting one change at a time.

The automatic generation of Modal diagrams frequently led to world boxes obscuring each other. Although, ideally, worlds would not be drawn overlapping each other, window sizes meant that this was sometimes necessary if an overview of the diagram was to be retained. A related issue is that the same world can be re-used to create a Modal diagram. If done unintentionally, this can confuse and clutter a diagram.

The original design of MoLE connected worlds with simple straight lines. Users commented that this implied a two-way connection to them, although the relationship was not actually symmetrical. It was suggested that arrowheads should be added. These would indicate direction; double-ended arrows can be used to denote symmetric relationships.

Users also expressed concern about the tree diagram used in the Universe window, which gave no indication of the pieces each world contained.

It's a shame that doesn't indicate it's got pieces in it as well.

(Subject 4)

The decision to separate the process and content elements was taken for two reasons. Firstly, section 3.8 identified that few subjects actually integrated the two representations. Secondly, pragmatic concerns over screen size, together with concerns of salience of information, suggest that simply providing names forms a sufficient link between the content and process representations. This link is then supported when appropriate, for example by identifying and offering examples relevant to claims which have been tested.

Unprompted, the subject retracted the request, using the second of these concerns as justification.

I guess it's the size issue. It would be cluttered, wouldn't it.

Implementation issues

MoLE proved to be extremely robust, crashing only twice during the tests. Indeed, one user expressed surprise that opening 30 windows had failed to crash the software.

Both the crashes were caused by the same problem: renaming a world to which other worlds were connected. When worlds were renamed, the UniverseView updated itself and in doing so tried to draw arrows to a world that was no longer recognised by the universe's 'worlds' dictionary. This problem was remedied by updating all 'connectionsList' variables prior to updating the View.

Due to variables being initialised, opening the "set up initial conditions" dialogue window invariably launched a warning dialogue saying that grid sizes could not be altered if worlds had already been created. The code was altered so that the warning was only given when the grid size was altered after worlds had been created.

Interface issues

All subjects encountered problems when using buttons to place worlds in the Universe window for the first time. Some also encountered the same problem when placing pieces on the grids for the first time.

The original mechanism required users to click once on the button, move the cursor over the View associated with the grid world or universe, and click again where the new object is to be placed. Confusion arose from the fact that pressing the buttons had no immediate effect.

The thing about buttons is that you always expect something to happen when you press them.
(Subject 4)

Although it was suggested that prompt windows could be used to show what was required, it was also commented that these would soon become irksome for users.

One subject commented that the buttons on the Universe window were in an odd order, suggesting that they be swapped around. This re-arrangement groups the buttons in the order in which they are used. It also places the most commonly used buttons on the right of the interface, above the sentence editing buttons, resulting in improved usability.

Two subjects encountered difficulties finding the menu command to create a diagram. The current design has this in the "file" menu of each grid world.

A final problem commented on by all of the subjects was the order in which sentences are built up. The system requires a buttons to be used in a fixed order: Modal operators first (if any are required), followed by a comparative predicate. Pressing one of these buttons launches a dialogue window, in which the pieces to be compared are entered.

A more natural way to enter sentences would be to choose the pieces to be compared, and then build up the predicate and operators around it. However, at least three extra buttons would be required to achieve this, and attempts to incorporate them led to a cramped and cluttered interface. In the interests of clarity, it was decided that this mechanism be left unaltered.

Platform-related issues

All users encountered difficulty using the X-windows environment supported by the workstation. The inclusion of a standard X-Windows menu in every window, in addition to the menu provided by MoLE, confused several users. Further problems were caused by the command 'close', which is equivalent to 'minimise' on a PC rather than 'close' on a PC or Macintosh. Further frustration was caused by the system's support for scrolling windows. Users found this to be far slower than they expected. For these reasons, the decision was taken to modify the tool for use on a Macintosh.

Summary

An evaluation of MoLE showed that it fulfilled its design aims, providing an environment in which concepts could be explored by creating and manipulating grid worlds, generating Modal diagrams, and testing Modal claims. Although the software was robust, several improvements to the design were identified.

5.5 The Redesign of MoLE

This section outlines the design refinements motivated by the results of the formative study described in section 5.3.

Provision of instructions

Having identified problems with the use of MoLE, a simple and potentially effective modification was to produce a sheet of instructions which address the concerns raised in this study. A guide to features, for example, will allow users to locate the "move pieces" command; a description of buttons will explain the order in which sentences are to be built.

Alterations to the class GridWorld

One subject encountered problems when the same piece was used to create two identical diagrams, one on top of the other. In order to avoid this, a new variable has been added to the class GridWorld. This variable is called 'hasSpawned'.

The variable keeps track of whether the world has been used to derive new worlds. Although the command to move pieces still works for such worlds, a warning dialogue is launched before it is carried out. This advises users against re-using the grid world unless they have already disposed of the previously created new worlds, and asks them to confirm whether or not they wish to proceed.

Additionally, new methods have been introduced which allow the generateWorlds method to be applied more than once. This allows diagrams several layers deep to be created with one simple action.

Alterations to the class GridWorldInterface

In response to problems encountered when trying to find the "move pieces" command, a new edit menu was added to the grid world window. This also contains the commands to move pieces more

than once, to wipe the grid clean, and to copy pieces in from another grid, allowing for the flexible creation and editing of grid worlds.

Alterations to the class UniverseController

Pressing buttons which seemed to have no immediate response was identified as being a problem for users. Since a dialogue window prompting users was considered to be too intrusive, a compromise was adopted whereby the cursor alters as it enters a view where, as a result of the button-click, an action is to be performed.

This has led to the development of the `controlInitialize` method for the class `UniverseController`. Normally, this changes the cursor from an arrow to a hand, to indicate that the window supports drag-and-drop. Now the method checks to see whether a world is to be placed or removed, and changes the cursor to a cross-hair or a 'ban' sign respectively (Figure 5.17).

The cross-hair cursor denotes that a new world is being 'carried', to be placed under the cross-hair when the mouse is clicked. When this occurs, the cursor reverts to being a hand. The ban sign works in the same way, except that it denotes that the user has chosen to delete a world.

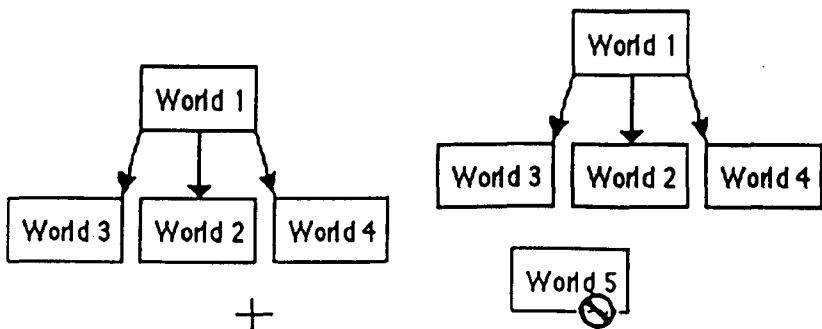


Figure 5.17: The additional cursors (cross-hair and 'ban') used by `UniverseController`

Alterations to the class UniverseView

A routine was added which allowed boxes to overlap, but not to completely obscure each other, addressing the problem of obscured worlds identified in the users' comments. Boxes which would otherwise completely overlap each other were stacked in a tiled pattern.

The original interface used plain lines to denote links between grid worlds. In response to comments suggesting that these were confusing, and implied a bi-directional relationship, the `displayOn` method was revised to use arrows instead.

Alterations to the class UniverseInterface

In response to comments about the order of the buttons in the `Universe` window, the interface was modified slightly. This resulted in the revised interface shown in Figure 5.18.

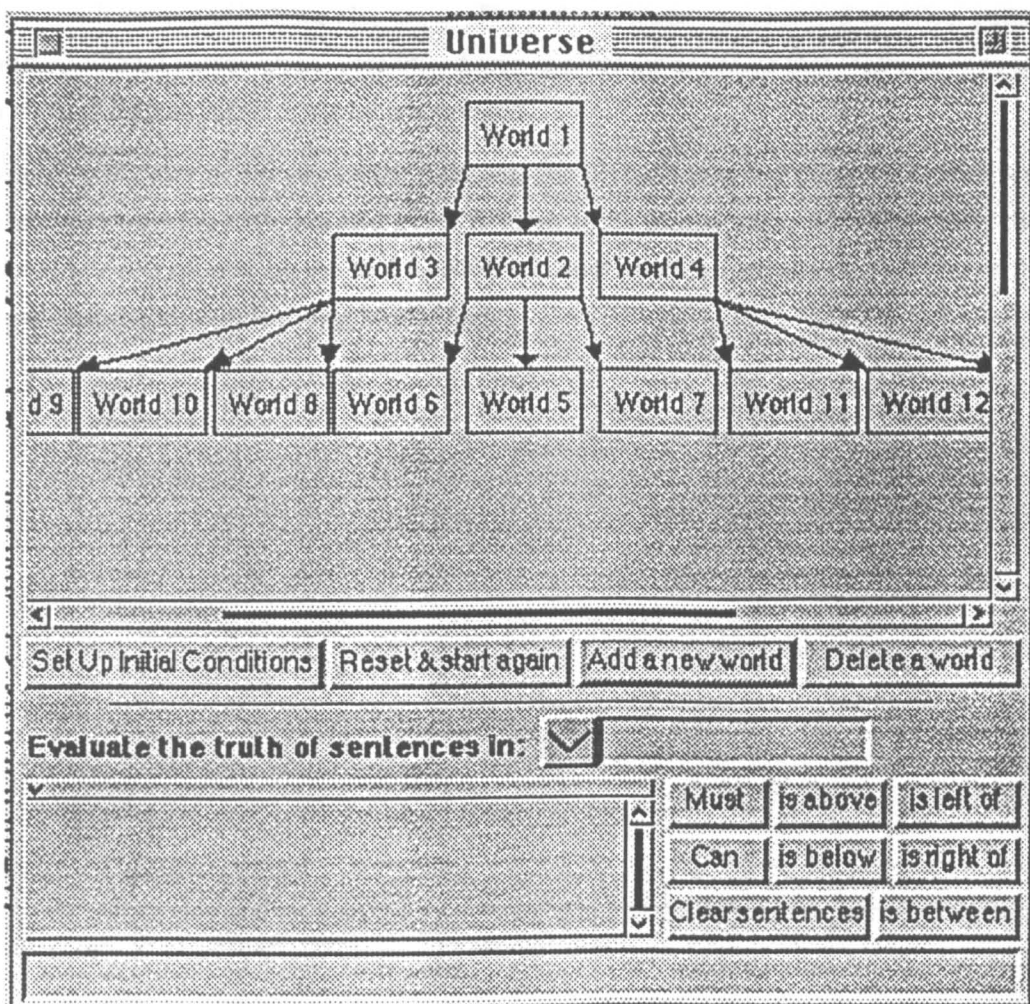


Figure 5.18: An example of a window supported by UniverseInterface

This differs from the original window design only in the order of the buttons used to place or delete worlds, to set up the initial conditions or to reset the universe.

5.6 Conclusions

This chapter has given an overview of the software tool, MoLE. To demonstrate the role of this tool in solving Modal logic problems, an example was given of how it can be used to investigate a Modal claim.

A formative evaluation of MoLE was carried out, which identified several design and implementation problems.

Whilst automatically generating diagrams proved to be simple and useful, it caused confusion when worlds were re-used as the starting point of a diagram, led to layout problems whereby one world completely obscured another, and, for diagrams more than one 'layer' deep, required users to undertake a trivial but laborious process of opening each new world in turn to extend the diagram. Extra functionality was added which addressed these issues, including warning messages, world detection routines, and the ability to generate more than one 'layer' at a time.

Pedagogically, comments led to the replacement of linking lines with arrows, which prevented confusion over whether or not links were intended to imply a two-way relationship.

Problems with the use of buttons also arose. The order of these was changed to reflect patterns of use, and additional cursor types were introduced to alleviate concerns that 'nothing happens' when buttons such as "place a world" are pressed.

Chapter six will describe a study which compares the performance of subjects using this revised tool and those working with pen and paper to learn about Modal logic. This will verify the additional benefits offered by the tool (section 4.2), and allow a further investigation of the problems associated with learning Modal logic.

Chapter 6

An Evaluative Study of MoLE

6.1 Introduction

Having demonstrated in chapter three that combining complementary representations provides effective support for learners, section 4.2 proposed that the effectiveness of these representations could be further enhanced by using them as the basis of a visualisation and manipulation tool. Using the representations in this way supports direct manipulation (section 2.5 argues that interactivity is a feature of good content representations), constrains the types of errors users could make, allows the software to offer examples which can help explain proofs, and automates rote procedures such as repeatedly drawing block worlds. These features reduce students' cognitive load, which should enhance learning (Sweller, 1988; 1989).

The development and implementation of MoLE, a software tool which incorporates these features, was described in chapter four. In chapter five, a formative evaluation of the tool was described, together with the revisions that resulted from this process.

This chapter describes an evaluative study of MoLE. The study has three aims: (i) to establish whether the software tool MoLE offers any advantages over traditional pen-and-paper teaching; in other words, whether the four features used to enhance the representation (interactivity, constraining errors, offering examples and automating rote tasks) really benefit the user; (ii) to investigate why certain topics in Modal logic are conceptually harder than others; and (iii) to identify further revisions which could usefully be made to MoLE. This study will also help answer the last four research questions identified in section 1.3, which concern the ways in which visualisation and manipulation tools support learning. These aims are achieved by comparing the performance and comments of two groups of learners, one working with MoLE, the other using pen and paper.

6.2 The Study

In order to test the effectiveness of MoLE, a study was carried out in which the performance of students using the software was compared with that of students working by hand. Participants were required to work through intervention material which concentrated on concepts identified in chapter 3 as being particularly difficult for learners. Students working with MoLE used the program to construct and work through examples in the teaching material; subjects in the pen and paper condition completed these by hand.

Since MoLE was developed to be a learning environment rather than a tutoring system, both instructional conditions used the same printed material. This ensured consistency between the groups in terms of the representations they were shown, suggesting that differences in performance between the instructional conditions should result from the additional features offered by MoLE.

Methodology

The methodology used in this study follows the format established by the study described in chapter 3. It combines an experimental methodology, used to establish factors influencing performance, with a qualitative analysis of responses to open questions in the post-test. In addition, to providing further insight into the differences between instructional conditions, qualitative data were sorted into categories according to topic (proof construction, system T, system S4, system S5, blocks-worlds, pros & cons of each instructional condition, and preferences for MoLE or working with pen and paper) in order to provide a deeper analysis of the problems associated with learning Modal logic. Further topics of importance were also identified, and these are reported on in the third section of results.

The remainder of this section describes the experimental design, together with the materials used to gather the qualitative data.

Hypotheses

It was hypothesised that subjects using MoLE would show greater pre- to post-test improvement on a variety of measures of conceptual improvement than subjects constructing proofs by hand. In addition, MoLE users were expected to progress further through the teaching material in the time allotted. These predictions are motivated by the additional features MoLE offers, such as the automation of routine tasks and support for direct manipulation.

Although MoLE uses complementary representations, it presents them in distinct windows, allowing the user to control the order in which they view the representations. By contrast, the diagrams in the teaching material show a unified view of the representations, with block worlds visible at each branching point on the tree diagram. This meant that MoLE users (who were required to use the printed material alongside the software) saw the representation in both formats, whilst their counterparts did not. It was predicted that giving users control over the use of complementary representations, and focusing on the process representation, would allow

subjects using MoLE to solve problems that included abstract claims as well as those than involved specific examples.

The ability of subjects in both instructional conditions to use examples (or counter-examples) to assess the validity of claims, and to transfer their understanding of modal proof construction to a new modal system (a near transfer of knowledge), was also investigated.

The study in chapter three suggested that there were differences between instructional conditions in subjects' ability to assess their performance. Since the design of MoLE was intended to promote deeper reflection on what was being learnt, it was predicted that MoLE users would be more accurate than their counterparts at assessing their performance.

Chapter three showed that prior experience with logic affected some measures of subjects' performance on key topics from Modal logic. Consequently, it was predicted that this would also have a significant effect on subjects' improvement in this study. Since the previous study found that gender only influenced the measure of the concept of reference (a topic excluded from this study), it was hypothesised that it would not affect any measures of performance. Additionally, using MoLE was predicted to be more motivating for subjects than completing the proofs using pen and paper.

Subjects

40 volunteers from computer science departments at 3 sites (Aberystwyth, University of Wales, Imperial college, London, and Queen Mary and Westfield college, London) took part in the study. Volunteers were paid for participation.

Materials

Subjects worked through three sets of material: a pre-test, intervention material, and a post-test. All materials were self-administered, and were developed from those used in the study described in chapter three. These were altered to concentrate on the more difficult concepts which had been investigated, specifically accessibility relations, proof construction, operator reduction problems, and the near-transfer of knowledge. All materials used in this study can be found in Appendix 8.

The pre-test gathered data on the subject's gender and prior experience with logic, and then set a series of questions designed to establish the subject's understanding of the topics to be covered in the study.

The intervention material introduced accessibility relations, proof construction, and the modal systems T, S4 and S5 (when to apply each system, and how the cancellation of modal operators varies from system to system) using a series of grid-world examples, such as the Modal diagram shown in Figure 6.1.

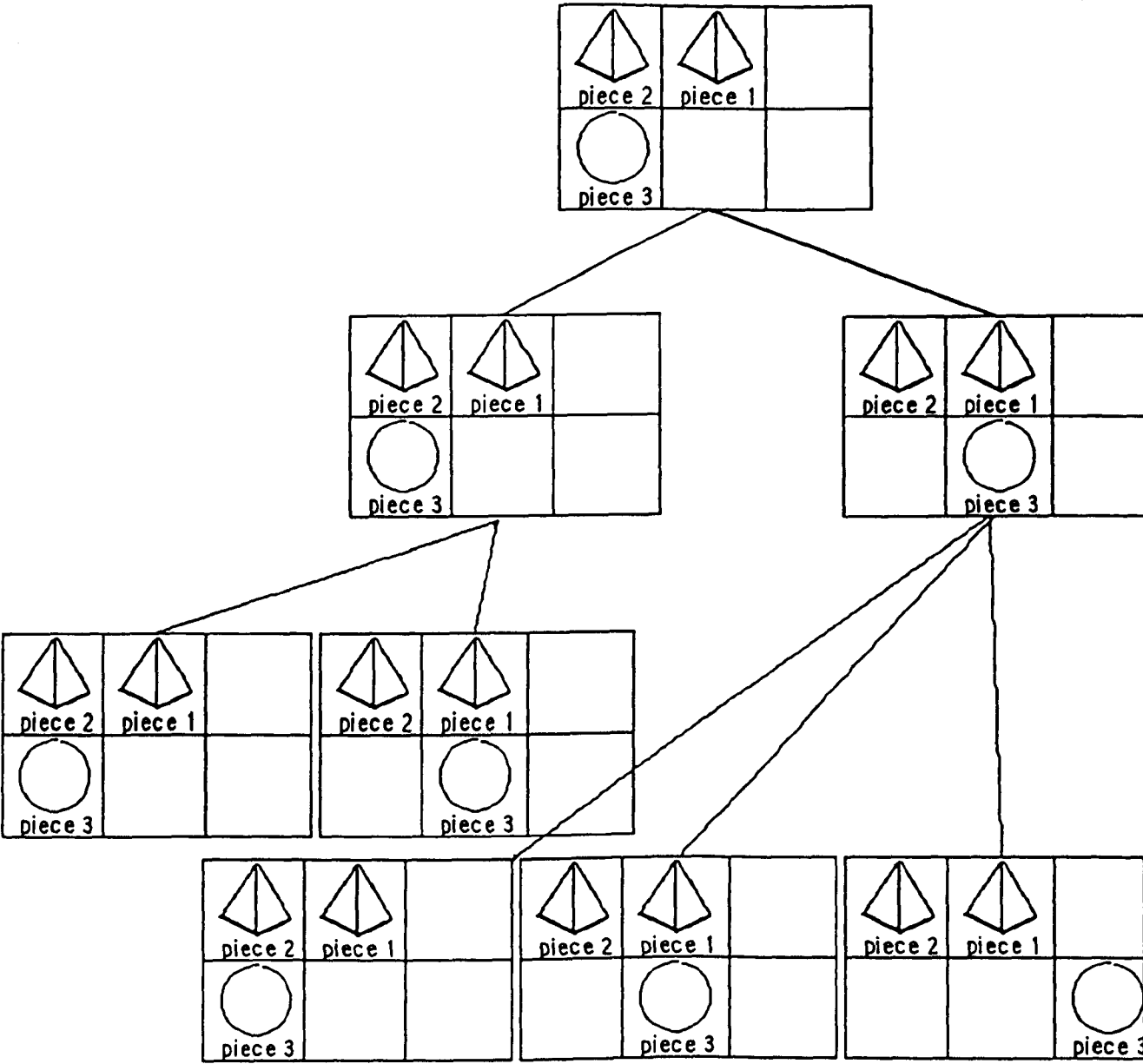


Figure 6.1: A Modal diagram

- Here, each rectangle represents a different 3x2 grid, each of which is a possible world. Three pieces have then been placed onto these grids (two tetrahedrons and a sphere). In order to create the diagram, the sphere are then allowed to move up to one square left, right, up or down, so long as this move leads to an empty square. Additionally, spheres are allowed to remain stationary, introducing an identity relationship. The lines connecting the worlds indicate that the lower grid has been derived from its predecessor by allowing spheres to move once.

In response to concerns raised in the first study (section 3.6), the material contained exercises which were intended to give subjects practice with using the concepts as they were introduced. Although performance on these exercises was not assessed, the number of questions attempted was noted, and was used to measure how far through the teaching material each subject had progressed.

Finally, subjects took a post-test, comprising of questions based on those in the pre-test, to assess improvement across the study. A second section of this test gathered qualitative feedback on the

learning process. Subjects were invited to say whether they had enjoyed taking part in the study, whether they thought they had done well or not, whether they had encountered problems understanding the concepts covered by the teaching material, if the blocks-world examples had been easy to understand, whether the teaching material contained an appropriate number of exercises, whether there were any advantages or disadvantages to the instructional condition they had been assigned to (either pen & paper or MoLE), and whether or not they would have preferred to work using the software. In order to inform the preferences, MoLE was demonstrated briefly for subjects working with pen and paper. Subjects who had used MoLE had already had to complete proofs by hand during the post-test.

Table 6.1 summarises which questions in the pre- and post-tests related to each of the concepts covered by the instructional material.

Table 6.1: questions and concepts in the pre- and post-tests

Concept	Question
Accessibility relations	1
Applicability of T, S4 and S5	2
Modal proof construction	3
Ability to use examples/counter-examples	4
Knowledge of appropriate reductions of modal operators (concrete examples)	5 a, b
Knowledge of appropriate reductions of modal operators (without examples)	5 c, d
Near-transfer of knowledge	6

Design

The study set out to investigate the improvement in several measures of conceptual performance, taking into account factors of instructional condition and prior experience with logic, using a randomised block experimental design.

Chapter 3 identified that prior experience with formal methods and notation significantly affected measures of performance for subjects in the learning environment condition. Consequently, in this study, subjects who had completed a first course in logic were classified as experts; those that had not are referred to as novices.

These considerations resulted in mixed design with a two level within-groups factor of time (test scores at pre- and post-test), and two two-level between groups factors: instructional condition and experience.

The dependent variables were measures of performance on the concepts identified above. Some of these produced category data, with subjects either being able to or else failing to demonstrate their understanding of a concept. Understanding accessibility relations, the ability to construct proofs, the ability to use examples, and completion of the near-transfer question all fell into this category. The remaining measures scored subjects on the number of Modal systems explained or used correctly.

It should be noted that equipment constraints led to slightly more subjects being assigned to the pen and paper condition than the software condition. However, care was taken to ensure that the distortion this caused was minimal.

Procedure

All subjects completed the pre-test, worked through the intervention material, and then attempted the post-test. Since the time taken to work through the intervention material was found to play a significant role in subjects’ improvement in the study described in chapter three (section 3.5), all subjects were given 50 minutes to work through the teaching material. This was based on the time 2 s.d.’s above the mean in the first study (55 minutes), which was modified to 1.7 s.d.’s in order to meet timetabling constraints.

6.3 Results: Pre- to Post-test measures

This section of results reports on measures taken at both pre- and post-test, in order to assess improvement on the different measures of performance. It precedes this with a brief overview of the distribution of subjects between the two instructional conditions. Full data relating to this section and to section 6.4 can be found in Appendix 9.

Distribution of subjects

Although subjects were assigned to instructional conditions using a randomised block design, allocation was influenced by equipment constraints. Table 6.2 summarises the distribution of subjects both by instructional condition and expertise. Additionally, the table breaks down the distribution by gender.

Table 6.2: Subject distribution by instructional condition, experience and gender

	Novices		Experts		Total
	Male	Female	Male	Female	
Pen & Paper	7	3	10	2	22
Computer	6	2	6	4	18
Total	13	5	16	6	40

Accessibility relations

Subjects were asked to explain accessibility relations in question 1; responses were then categorised as being correct or incorrect. Table 6.3 summarises the number of correct responses at pre- and post-test.

Table 6.3: Number of correct explanations of accessibility relations

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Pre-test	2	0	0	0
Post-test	8	6	6	7
Total in category	10	12	8	10

Chi-squared tests were carried out to see if the data differed by instructional condition or expertise. No significant differences were found. McNemar tests confirmed that both experts and novices in each instructional condition improved significantly from pre- to post-test (pen & paper, novices, $p<0.0313$; pen & paper, experts, $p<0.0313$; computer, novice, $p<0.0313$; computer, expert, $p<0.0156$).

Modal systems

Question 2 required subjects to explain the three Modal systems. This question was scored from 0 to 3 according to how many systems they successfully explained. Table 6.4 summarises the means and variances by instructional condition and expertise.

Table 6.4: Means and variances of the number of systems correctly explained

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Pre-test	0 (0)	0 (0)	0 (0)	0 (0)
Post-test	1.70 (1.57)	1.42 (1.90)	2.38 (1.12)	2.20 (1.73)

No-one successfully answered any of the questions on this topic at pre-test. Because of this, comparisons of pre- to post-test data fail homogeneity of variance tests. Consequently, the data were analysed using an [2x2] ANOVA on the post-test measures alone, with between groups factors of instructional condition and expertise. This showed a trend for an effect of instructional condition ($F(1,39)=3.206$, $p<0.08$), with MoLE users performing better than their counterparts. There were no significant effects.

Proof construction

In question 3, subjects were asked to construct a proof of a Modal claim. Answers were categorised according to whether or not subjects completed a diagram which validated the claim under investigation. Answers which only used natural language, but which clearly described the conditions required for the proof to hold, were also accepted. Results are summarised in Table 6.5.

Table 6.5: Number of correctly constructed diagrams

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Pre-test	0	0	0	0
Post-test	7	6	6	8
Total in category	10	12	8	10

Chi-squared tests were carried out to see if the data differed by instructional condition or expertise. No significant differences were found. McNemar tests confirmed that both experts and novices in each instructional condition improved significantly from pre- to post-test (pen & paper, novices, $p<0.0156$; pen & paper, experts, $p<0.0313$; computer, novice, $p<0.0313$; computer, expert, $p<0.0078$).

Use of examples and/or counter-examples

Question 4 required subjects to give an example which demonstrated the validity of a claim (or a counter-example which refuted it). Responses were categorised according to whether subjects managed to do this. Table 6.6 shows how many subjects used examples or counter-examples correctly, categorised by instructional condition and expertise.

Table 6.6: Number of subjects able to use examples correctly

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Pre-test	0	0	0	0
Post-test	4	6	5	7
Total in category	10	12	8	10

The post-test data were analysed by instructional condition and expertise using chi-squared tests. No significant differences were found. Interestingly, McNemar tests showed that experts made significant pre- to post-test improvements, whilst novices did not; however, there was a trend for novices using MoLE to learn how to use examples (pen & paper, novices, $p < 0.125$; pen & paper, experts, $p < 0.0313$; computer, novice, $p < 0.0625$; computer, expert, $p < 0.0156$).

Application of systems (operator reduction)

Question 5 asked subjects to say under which conditions certain cancellations of operators were valid. The question comprised of 4 examples (a-d), each of which required subjects to indicate if the reduction was valid in T, S4 or S5. Answers were scored according to the number of systems correctly given, together with the number of systems correctly omitted. For example, if a reduction was S4 valid, an answer which just said "S4" would score 2; 1 for correctly indicating that it is S4-valid, and 1 for omitting T (since the decomposition is not T-valid). Since all S4-valid decompositions are also S5-valid, the full answer (scoring all 3 points) should be, "S4, S5". This resulted in scores between 0 and 12. Table 6.7 summarises the means and variances for responses to this question by instructional condition and expertise.

Table 6.7: Means and variances of scores on the operator reduction system

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Pre-test	0 (0)	0 (0)	0 (0)	0 (0)
Post-test	1.80 (9.96)	1.83 (7.97)	4.88 (16.98)	6.30 (19.34)

As indicated in the table, there were no correct answers recorded at pre-test for any subjects. As a result, comparisons of pre- to post-test data fail homogeneity of variance tests. Consequently, the data were analysed using an [2x2] ANOVA on the post-test measures alone, with between groups factors of instructional condition and expertise. There was no effect of expertise, although there was a main effect of instructional condition ($F(1,39)=10.664$, $p < 0.002$), with MoLE users performing better than subjects working with pen and paper. No significant interaction between instructional condition and expertise was observed.

Additionally, question 5 sought to investigate whether subjects were able to deal equally well with both concrete and more abstract questions. The reduction of Modal operators is not affected by the statement which the operators precede; in other words, asking if $MMCMMCCMp$ can be reduced to $MCMp$ is not dependent upon what 'p' denotes. One indication of the depth of a subject's understanding of the topic is whether they are aware of this generalisation, or whether their appreciation is limited to specific examples. To explore this, question 5 included both parts with examples (referred to as 'concrete') and parts without (referred to as 'abstract'). Parts (a) and (b) of question 5 provided a blocks-world example as part of the question, and so were classified as being concrete questions. Parts (c) and (d) did not include an example, and so were

classified as being more abstract. Table 6.8 shows the number of subjects who attempted the different types of questions.

Table 6.8: Number of subjects who attempted concrete and abstract questions

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
No questions	5	6	2	2
Only concrete	3	2	3	1
Both	2	4	3	7

This information was then transformed, with subjects scoring 0 if they had attempted to answer none of the parts of question 5, 1 if they had attempted either (a) or (b), and a further 1 point if they had attempted the more abstract questions given in parts (c) and (d). This gave a range of scores from 0-2. Table 6.9 summarises the means and variances by instructional condition and expertise.

Table 6.9: Means and variances of scores for the types of reduction questions attempted

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Post-test	0.70 (0.68)	0.83 (0.88)	1.13 (0.70)	1.50 (0.85)

Values were compared by an [2x2] ANOVA, with between groups factors of instructional condition and expertise. This revealed a trend for an effect of instructional condition ($F(1,39)=3.871$, $p<0.057$), with computer users performing better than their counterparts, but no significant results.

Given that more MoLE users managed to attempt the abstract operator reduction questions, the question then arises as to whether this difference accounts for their greater overall score. In order to investigate this, question 5 was re-scored, only taking into account answers to parts (a) and (b). This revised score allows an investigation into whether subjects in the two instructional conditions were equally good at concrete operator reduction questions. Table 6.10 summarises the means and variances for this new score by instructional condition.

Table 6.10: Means and variances of scores for concrete operator reduction questions

	Pen & Paper	Computer
Post-test	1.23 (3.04)	3.33 (4.94)

These means were compared using an ANOVA with a between groups factor of instructional condition. This revealed a significant effect of instructional condition ($F(1,39)=11.285$, $p<0.0018$), with MoLE users performing better than subjects who had worked with pen and paper. This indicates that MoLE users were better at concrete questions, as well as being better overall; in other words, simply being better at abstract questions fails to account for all of the difference between the groups.

Near transfer

Subjects' ability to transfer what they had learnt to a new system was assessed in question 6. Subjects were shown a simple Modal system, not previously encountered, and asked to demonstrate (by use of an example) whether or not a claim was valid.

No answers were given for this question at pre-test. Table 6.11 shows how subjects performed at post-test.

Table 6.11: Number of subjects who attempted the near transfer question

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Did not attempt	7	9	5	4
Incorrect answer	0	0	1	1
Correct answer	3	3	2	5

The post-test data were analysed using chi-squared tests. No significant differences were found. McNemar tests showed that only experts using the computer showed a trend for a pre- to post-test increase in ability to transfer knowledge to a new Modal system ($p < 0.0625$).

Summary

Measures for the concepts of accessibility relations, Modal systems, Modal proofs, the use of examples and counter-examples, and the near transfer of knowledge were tested at pre- and post-test. Instructional condition was found to affect users' ability on all measures of performance involving Modal systems. Expertise only affected on measure: pre- to post-test increases in the number of people able to use examples or counter-examples when solving problems. There was also a trend for experts using MoLE to learn how to transfer knowledge to a new Modal system.

6.4 Results: Post-test measures

This section of results reports on measures taken at post-test only. These were primarily self-assessed questions relating to motivation or understanding. The section also includes analyses based on the number of exercises subjects attempted during the intervention material.

Subjects' understanding of Modal systems

Subjects were asked to say whether they fully understood, could apply but not understand, or failed to understand each of the three Modal systems presented in the instructional material. Responses were analysed using Chi-squared tests, categorised first by instructional condition and then by expertise. Table 6.12 summarises this.

Table 6.12: Subjects' self-assessed understanding of systems T, S4 and S5

		Pen & Paper		Computer	
		Novice	Expert	Novice	Expert
System T	Didn't understand	0	2	1	0
	Apply only	3	3	2	1
	Understood	7	7	5	9
System S4	Didn't understand	0	2	1	1
	Apply only	6	6	3	2
	Understood	4	4	4	7
System S5	Didn't understand	4	7	4	2
	Apply only	5	3	2	3
	Understood	1	2	2	5

For systems T and S4, almost all subjects felt that they knew enough to apply the system. As a result, responses were categorised according to whether subjects fully understood the system or not. No significant differences were found between subjects' self-perceived understanding of the systems, by either instructional condition or expertise, although there was a trend for subjects using MoLE to understand S5 better than subjects using pen and paper (1 d.f., $p < 0.10$).

It was hypothesised that subjects using MoLE would be better at assessing their own performance than those working with pen and paper. To investigate this, subjects' self-assessed understanding of Modal systems was correlated with their actual performance, to see if their assessment was accurate. There was a strong trend for correlation in the computer condition (Spearman's 2-tailed, $r=0.4665$, $p<0.051$), but no equivalent correlation for the pen and paper condition (Spearman's 2-tailed, $r=0.1878$, $p<0.403$). This suggests that MoLE users were able to assess their understanding of Modal systems more accurately than their counterparts.

Subjects' understanding of proof construction

Subjects were also asked to indicate whether or not they had encountered problems constructing proofs. Table 6.13 shows the responses, categorised by instructional condition and expertise.

Table 6.13: Subjects' assessment of problems constructing proofs

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Had difficulty	6	7	3	2
No difficulty	4	5	5	8

Results were analysed using Chi-squared tests, categorising data first by instructional condition and then expertise. This revealed a significant difference between instructional conditions, with those working with pen and paper reporting significantly more problems (1 d.f., $p<0.05$). No differences were found for levels of expertise.

Subjects were then classified by whether or not their perceived difficulty accurately reflected their performance on question 3. This information is presented in Table 6.14.

Table 6.14: Subjects' accuracy at assessing whether they had accurately constructed a proof

		Pen & Paper		Computer	
		Novice	Expert	Novice	Expert
Accurately assessed ability	Constructed	2	3	4	7
	Failed	1	4	1	1
Inaccurately assessed ability	Constructed	5	3	2	1
	Failed	2	2	1	1

This information was analysed using a series of Chi-squared tests, to assess whether accuracy was affected by ability, instructional condition or experience. Neither expertise (prior experience with formal logic) nor having successfully constructed a proof (actual ability to use this skill) were found to affect assessment of ability. There was, however, a trend for subjects using MoLE to be more accurate in their assessment of their ability to construct proofs (1 d.f., $p<0.10$).

Motivation

Subjects were asked to indicate whether they had enjoyed taking part in the study, whether they thought it was 'o.k.', or whether they had not enjoyed it. This was scored from 1 to 3 (with 3 as enjoyable). Table 6.15 summarises the means and variances of the data, categorised by instructional condition and expertise.

Table 6.15: Means and variances of subjects' enjoyment

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Post-test	2.40 (0.27)	2.67 (0.42)	2.25 (0.21)	2.70 (0.23)

Results were analysed using an [2x2] ANOVA, with between groups factors of instructional condition and expertise. There was found to be a main effect of expertise ($F(1,39)=4.245$, $p<0.047$), but no effect for instructional condition, or for the interaction of instructional condition and expertise.

Similarly, subjects were asked to indicate how well they thought they had performed. These data were also scored from 1 to 3, and the means and variances are presented in Table 6.16.

Table 6.16: Means and variances of subjects' perceived improvement

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Post-test	2.20 (0.42)	1.92 (0.67)	2.50 (0.76)	2.50 (0.53)

An [2x2] ANOVA was carried out on the data, revealing a main effect of instructional condition ($F(1,39)=5.187$, $p<0.029$), which indicated that MoLE users were more confident that they had improved. No effects for expertise or the interaction between instructional condition and expertise were found.

It was hypothesised that MoLE users would be better at assessing their ability than subjects using pen and paper. To investigate this hypothesis further, subjects' confidence in their performance was correlated with their actual performance, to see if their assessment was accurate. There was significant correlation for MoLE users (Spearman's 2-tailed, $r=0.6824$, $p<0.001$), and a slight trend for correlation for the pen and paper condition (Spearman's 2-tailed, $r=0.4160$, $p<0.061$). Both groups seem able to assess their performance, although this can be said with greater confidence for those in the MoLE condition.

Comments about the instructional conditions

Finally, subjects were asked to say if there was anything particularly good or particularly bad about the instructional conditions they had been assigned to. Table 6.17 shows the pattern of subjects' responses. It should be noted that the totals are not the same as for other tables, as some subjects did not express an opinion, and others identified both pros and cons for their instructional condition.

Table 6.17: subjects' opinions about the pros and cons of their instructional condition, and the total number of subjects in each of the categories, for comparison

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Anything particularly good?	3	6	8	8
Anything particularly bad?	8	5	1	4
Total in category:	10	12	8	10

A series of Chi-squared tests were carried out, grouping responses to each of the questions both by expertise and by instructional condition. There were no differences by expertise; however, significantly more subjects using MoLE than using pen and paper gave an example of something particularly good about their instructional condition (1 d.f., $p<0.005$). Also, significantly more

subjects using pen and paper than using MoLE found gave an example of something particularly bad about their instructional condition (1 d.f., $p<0.05$).

Number of exercises attempted

A record was taken of the number of questions each subject attempted whilst working through the intervention material. Note that these Figures do not distinguish between correctly and incorrectly completed questions, since this measure is only concerned with how far through the material the subjects had progressed. Table 6.18 summarises the means and variances of the number of questions attempted, categorised by instructional condition and expertise.

Table 6.18: Means and variances of the number of exercises attempted

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Post-test	7.80 (3.73)	7.00 (6.55)	9.63 (5.18)	10.50 (24.48)

Values were compared using an [2x2] ANOVA, with between groups factors of instructional condition and expertise. This revealed a significant main effect of instructional condition ($F(1,39)=4.883$, $p<0.034$), with MoLE users completing more exercises. There was no effect of expertise, nor of the interaction between instructional condition and expertise.

This raised the question of whether differences in scores were actually the result of MoLE being more effective than pen and paper for teaching Modal logic, or whether they were simply due to the fact that subjects who used MoLE covered more of the concepts which were assessed than those who did not. In order to ascertain which of these was the case, responses to question 5 (operator reduction) were re-scored, only considering marks for systems T and S4. Question 5 was chosen for this analysis, since topics from across the teaching material were required to answer it fully. Marks were restricted to the first two systems because there were no significant differences between the number of subjects in each group who had reached this point in the intervention material (Chi-squared, 1 d.f., $p>0.10$, based on measures of progress through the exercises). The means and variances for this re-scored measure, categorised by instructional condition and expertise, are presented in Table 6.19.

Table 6.19: Means and variances for the re-scored measure of ability to apply Modal operators to cancellation problems

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Post-test	1.30 (6.46)	1.50 (4.64)	3.13 (7.55)	4.10 (8.77)

An [2x2] ANOVA with between groups factors of instructional condition and expertise showed a significant main effect for instructional condition ($F(1,39)=47.953$, $p<0.011$), with subjects using MoLE performing better than subjects using pen and paper, but no effect for expertise or the interaction between expertise and instructional condition. This confirmed that MoLE users not only covered more material, but were also better on performance measures restricted to material both groups had covered.

Other Effects

Measures of performance were also tested to see if there was any effect of gender. Two measures were found to be affected. The first was subjects' enjoyment of taking part in the study, which

demonstrated a slight trend for gender ($F(1,39)=3.333, p<0.085$), with males having enjoyed taking part more than females. There was no interaction between gender and instructional condition. The means and variances of scores for enjoyment, categorised by instructional condition and gender, are presented in Table 6.20.

Table 6.20: Means and variances of scores for enjoyment, by instructional condition and gender.

	Pen & Paper		Computer	
	Male	Female	Male	Female
Post-test	2.64 (0.49)	2.20 (0.70)	2.58 (0.27)	2.33 (0.27)

The second measure which showed an effect of gender was the revised scoring of Modal systems (restricted to systems T and S4) described above. Table 6.21 summarises the means and variances by instructional condition and gender.

Table 6.21: Means and variances for the re-scored measure of ability to apply Modal operators to cancellation problems

	Pen & Paper		Computer	
	Male	Female	Male	Female
Post-test	1.65 (6.12)	0.60 (1.80)	4.50 (6.27)	2.65 (0.24)

Results indicated main effects of instructional condition ($F(1,39)=35.548, p<0.02$) and of gender ($F(1,39)=4.120, p<0.05$), with subjects using MoLE scoring higher than subjects working with pen and paper, and males scoring higher than females. There was no significant interaction between the two factors. No obvious explanation for the difference in means between genders was found in the comments gathered at post-test.

Measures were also tested by site. Only one measure was found to be affected: subjects' confidence in their performance ($F(2,39)=6.5471, p<0.0037$). Tukey tests indicated that subjects at Aberystwyth and Imperial were more confident than subjects at Queen Mary and Westfield ($p<0.05$). The means and variances on this score, which ranged from 1 (believed they had done badly) to 3 (believed they had done well), are presented in Table 6.22.

Table 6.22: Means and variances of scores for confidence, by site.

	Aberystwyth	Imperial	Queen Mary and Westfield
Confidence	2.41 (0.26)	2.46 (0.27)	1.70 (0.46)

This factor is likely to result from the equipment constraints, which resulted in most of the students at Queen Mary and Westfield being assigned to the pen and paper condition. Instructional condition has already been shown to influence confidence.

Summary

Instructional condition influenced almost every measure of motivation and understanding. MoLE users professed to having greater understanding of proof construction, understood the Modal system S5 better, and were more confident of what they had learnt about Modal logic. They were also more accurate in their assessment of their ability, and more likely to find something positive to say about their instructional condition.

Measures of how many exercises each subject had attempted during the intervention also showed an effect of instructional condition, with MoLE users having completed more exercises than their counterparts.

Enjoyment of the study was found to be affected by expertise; there was also a slight trend for men to have enjoyed taking part more than women.

6.5 Results: Qualitative data

In addition to the measures described above, qualitative data was gathered which addressed several specific concerns, mainly motivated by the study described in chapter three. Subjects were invited to comment on whether they had found the blocks-world metaphor easy to understand, whether they felt the number of exercises they were asked to attempt during the intervention material was reasonable, and (after being shown both alternatives) which instructional condition they would have preferred to work in. Additionally, they were asked to comment if they had encountered any problems with the concepts identified as being difficult by the first study (the systems T, S4 and S5, and constructing Modal proofs).

In addition to investigating these topics, analysis of the data revealed five additional concerns shared by subjects:

- i) The effort involved in completing routine tasks
- ii) Time pressures
- iii) Deep understanding and reflection
- iv) Difficulties with abstract concepts
- v) Expectations and criticisms of the software

This section outlines responses to the topics mentioned above, and details the additional issues which were identified. Comments are collated in Appendix 10.

Assessment of the representation

Subjects were asked whether the blocks-world examples were easy to understand, whether they thought there were enough, too few or too many exercises, and whether they would have preferred to work with pen and paper or MoLE. Table 6.23 presents their responses.

Table 6.23: Subjects’ responses to questions about blocks-worlds, the number of exercises, and their preference for instructional condition

		Pen & Paper		Computer	
		Novice	Expert	Novice	Expert
Block examples easily understood	Yes	10	12	6	9
	No	0	0	2	1
Number of exercises	Too many	2	0	1	0
	About right	7	10	6	8
	Too few	1	2	1	2
Would have preferred to use...	Pen & paper	1	3	1	3
	MoLE	9	9	7	7

In total, 92.5% of subjects thought that blocks-world examples were easy to understand, 77.5% felt that the number of exercises set in the teaching material was ‘about right’ (with an additional 15% feeling that there were too few), and 80% of subjects saying that they would have preferred to use MoLE than pen and paper. This supports the motivational benefits of

MoLE already identified, and confirms that the illustrative 'content' representation provided by the blocks-world is effective for this topic.

Problems with Modal systems

System T, the first of the Modal systems to be introduced, appears to have been quite simple; only six subjects commented that they had encountered problems with this topic. Most of these comments expressed concern about not having been able to spend enough time on the topic; the remainder were less specific, simply saying things like,

I never used system T in logic, therefore I did not know from where to start.

(Subject 12; pen & paper)

System S4 is slightly more interesting. Although only nine subjects said that they had found this problematic, eight of them (89%) were from the pen and paper condition. Again, time pressures were the most frequently mentioned difficulty. Two other problems were also identified: applying knowledge about operator reduction and not having understood T. Since system S4 is an extension of system T, having had problems with this first system blocks any further development of subjects' understanding, making it impossible for them to learn more advanced concepts such as systems S4 and S5.

In total, 16 subjects commented that they had found S5 difficult. Of these, 10 were concerned with time pressures, and four with not having understood the previous systems. One other comment indicates that the subject found the theory behind operator reduction in S5 difficult to follow. Again, there was an imbalance between the instructional conditions in the number of comments, with 75% being from subjects in the pen and paper condition.

These comments confirm that Modal systems are conceptually difficult for subjects. They also highlight the hierarchical nature of Modal systems, with subjects needing to master each in turn before progressing. More importantly, however, the imbalance in comments between instructional conditions and the predominance of time-related concerns emphasises that MoLE is a useful tool to address these problems, since it has been shown to help speed progress.

Problems with Proof Construction

47.5% of subjects commented that they had encountered problems with constructing Modal proofs. As noted above, although both groups were equally good at this procedure, MoLE users were better at assessing their performance. As a result, only 38.8% of MoLE users commented on having difficulty, compared to 54.5% of subjects in the pen and paper condition.

Problems included the order of evaluating Modal operators (should they be read left to right, or from 'inside' to 'outside'?) and the problem of interpreting what a string of Modal operators might mean. There were also comments indicating that subjects were unsure about what constituted a proof.

Wasn't sure when to draw pictures or up to which point drawing consisted of a proof.

(Subject 9; pen & paper)

One subject commented that providing more examples would have helped address these problems. This subject was from the pen and paper condition.

These comments indicate that subjects encountered both procedural and conceptual problems relating to proof construction. Re-classifying comments according to whether they described procedural, conceptual, or other problems (e.g. time pressures) revealed that roughly the same number of subjects in each instructional condition had procedural and 'other' problems. Interestingly, seven subjects in the pen and paper condition had conceptual difficulties, compared with only three in the computer condition. This suggests that MoLE is particularly effective at addressing the conceptual problems associated with proof construction, possibly because MoLE users were able to complete more exercises than their counterparts. This ties in with the concern raised by the subject in the pen and paper condition that giving more examples supports concept formation.

Further issues

In addition to assessing the representation and discussing problems associated with Modal systems and proof construction, five further topics were identified which had not been encountered in the study described in chapter three. These were: the effort involved in completing routine tasks, time pressures, deep understanding, difficulties with abstract concepts, and software issues.

(i) Effort

50 of the 138 comments gathered on the post-test related to the effort involved in completing routine tasks. These comments covered, for example, drawing diagrams,

Drawing the modal diagrams could be a bit tedious after two or three layers.
(Subject 10; pen & paper)

the problems related to parsing the diagrams in order to evaluate claims,

It saves a lot of time computing for all the worlds.
(Subject 33; MoLE)

and the role subjects felt the software performed (or ought to perform) in reducing this.

Easy to construct the diagrams and view the diagrams to solve the problems.
(Subject 37; MoLE)

Clearly, this was a topic of key importance to subjects. Many of the comments referred to the way this effort hindered learning.

If the examples can be done more quickly, then more time can be spent learning new material.
(Subject 5; pen & paper)

Other comments suggested that software could help redress this.

You don't have to draw lots of diagrams, which are messy and make it more difficult to understand the essential principles... You can concentrate on actual material rather than messy diagrams.
(Subject 32; MoLE)

These findings support the design criteria for MoLE, which emphasised the importance of automating routine tasks. Considering only the comments which identify effort (rather than considerations of what the software ought to do or advantages of the other instructional condition) gives further support to this. Subjects in the pen and paper condition produced 17 comments about the difficulties they had encountered working through the exercises; by contrast, only 1 MoLE user (Subject 32) expressed such a concern. This was that "it is sometimes easier to do it by hand", and was a qualification of the preceding comment that software is "easier

sometimes, you can concentrate on actual material rather than messy diagrams.” This imbalance between the instructional conditions indicates that MoLE addressed the issue of dealing with rote tasks effectively.

(ii) Time pressures

A large proportion of the comments gathered (26%) related to time pressures. This category includes comments such as,

I don't feel I had enough time to go through everything.

(Subject 21; pen & paper)

quotes indicating that certain processes were particularly time consuming,

Spend too much time on drawing grid worlds and symbols.

(Subject 5; pen & paper)

and responses expressing the belief that the other instructional condition (i.e. working with MoLE for those that had been using pen and paper, and working with pen and paper for those who had used MoLE) would allow the subject to progress through the teaching material more quickly.

Would have spent less time drawing diagrams, and therefore have more time to spend answering questions and learning the concepts.

(Subject 10; pen & paper)

Table 6.24 shows how many subjects made such comments, grouping by instructional condition and expertise.

Table 6.24: Number of subjects expressing concern at time pressures

	Pen & Paper		Computer	
	Novice	Expert	Novice	Expert
Commented	7	9	5	2

Chi-squared tests were used to investigate differences between instructional conditions and levels of expertise, comparing the numbers of subjects who made time pressure-related comments to those who did not. Significantly fewer subjects using MoLE made such comments (1 d.f., $p < 0.05$). There was no significant difference for expertise. This supports the finding that MoLE users were able to work through the teaching material faster than subjects constructing proofs by hand (section 6.4).

(iii) Deep understanding and reflection

Although it was mentioned less often than time or effort, a frequently recurring concern was the depth of learning that each instructional condition supported (17.4% of comments). Feelings on this were mixed, with comments arguing in favour of each of the instructional conditions.

When you do it for yourself on pen & paper, you're made to actually consider the problem properly.

(Subject 35; MoLE)

You don't have to draw lots of diagrams, which are messy and make it more difficult to understand the essential principles.

(Subject 32; MoLE)

Comments both for and against each instructional condition were spread fairly evenly by instructional condition and gender. However, expertise clearly accentuated concerns, with 75% of comments in this category coming from subjects classified as experts.

Because the comments are quite varied and contain few illustrations as examples, only two approaches argued to promote deeper understanding could be identified. The first relates to diverting time from routine tasks to reflection (covered in the previous two sections). The second requires subjects to take a more active role in the construction of diagrams. This includes the argument that requiring subjects to draw proofs by hand prevents them from relying on the computer and so ignoring gaps in their understanding.

When you don't understand it properly before, you [won't] get any benefit from it as pressing button on the computer doesn't guarantee you know what you are doing.

(Subject 33; MoLE)

The key problem with this second theme is that it conflicts with the comments relating to effort. Specifically, these comments suggests that drawing diagrams will promote reflection and deeper understanding. The comments already covered argue that these same actions add to the existing cognitive load of trying to understand concepts, effectively *reducing* the capacity for higher-level learning.

No supporting evidence for any benefit to using pen and paper was found in this study. Whilst the experimental findings clearly indicate several performance-related benefits to using MoLE, it does not necessarily follow that this reflects deeper learning. Nonetheless, several measures (ability to accurately assess performance, ability to apply Modal systems to abstract operator reduction problems, etc.) *do* imply a deeper level of awareness amongst MoLE users. These results, taken with the importance of reducing effort, suggest that software tools such as MoLE can be more effective at promoting deeper understanding than working with pen and paper.

(iv) Abstraction

Another problem which received comment was the difficulty experienced in dealing with abstract or theoretical concepts, such as operator reduction or the nature of proofs.

High complexity of trying to conceptualise some of the accessibility relations (e.g. MCMCMCM etc).

(Subject 16; pen & paper)

Whilst the number of comments was not, of itself, dramatic (12 in total), the distribution was interesting. 11 of the 12 comments were from subjects working with pen and paper (3 from 2 novices, 8 from 3 experts), with only one comment from a subject using MoLE (a novice).

No direct explanations were apparent from the comments themselves. However, indirect explanations can be drawn from the comments relating to levels of understanding, such as,

You can concentrate on actual material rather than messy diagrams.

(Subject 32; MoLE)

Interaction imprints information on the memory.

(Subject 38; MoLE)

[MoLE] made it easier to understand because of the graphical representation.

(Subject 40; MoLE)

These last two comments suggest that part of the benefit of MoLE may have been the ability to interact directly with the representation, reducing the *perceived* level of abstraction. In terms of Green's cognitive dimensions (1989, 1994), both instructional conditions ought to have the same level of abstraction, since they make use of the same representation. If anything, MoLE's level of

abstraction should be higher than that of the pen and paper condition, since it separates the grid world examples from the diagram encoding the relationship between the worlds. This decoupling means that each process diagram can encode the structure of any number of examples.

This mismatch between the perceived and actual levels of abstraction is particularly interesting. It suggests that cognitive dimensions may need to take users' perceptions into account when discussing the effectiveness of a notational system.

Another possible explanation for why subjects using MoLE had fewer problems with theoretical and abstract concepts is that they only had to input claims using fragments of natural language, whilst those working with pen and paper used the less familiar symbolic equivalents. Use of natural language may have acted as a bridge between the familiar and the abstract. However, this explanation should be seen as additional to, rather than replacing, the other; alone, it fails to account for the comments indicating a difference between the perceived and the actual levels of abstraction, or the comments concerning visualisation and manipulation.

(v) Software issues

Expectations about the features a software tool ought to offer, comments from users of MoLE which relate to these expectations, and criticisms of MoLE, account for 25% of post-test comments. These have been grouped into seven themes.

The largest number of comments (8) related to the importance of the software's ability to automate routine tasks, and the effects this would have. MoLE already constructs diagrams, addressing most of the comments raised. However, an additional design-related concern was identified:

Not easy to extend diagrams — had to open lots of worlds.

(Subject 34; MoLE)

This comment related to the problem of extending the diagram, once several worlds are in place. Presently, this involves opening up each world, and asking each in turn to move the pieces on the board. A simple alternative would be to introduce a method which searched for worlds without connections, and then extended the diagram from each of these points. This would also reduce the system's viscosity, and leave the increased level of abstraction (distinguishing between 'extended' and 'unextended' worlds) to the software. This should make the system more effective for learners (Green, 1989).

Three comments were grouped to form the second theme in this category: that visualisation and manipulation tools can promote understanding. However, these comments were comparatively vague; only 'interaction' was mentioned as a way in which these benefits could be achieved.

The second largest grouping of comments (7) discuss issues of clarity in the diagrams. Subjects expected MoLE to produce neater diagrams than they would manage by hand.

It would have been less messy.

(Subject 8; pen & paper)

However, one subject who used MoLE remained unsatisfied with the diagrams it produced.

It was slow, and the diagrams of the world were messy which was a major problem.

(Subject 31; MoLE)

This issue is problematic. Whilst it would be comparatively simple to prevent overlap when drawing the structure-encoding diagram, there are practical constraints to consider, such as the number of worlds to be drawn and the screen size available. Nonetheless, this criticism does identify an opportunity for improving the current system.

The same subject also raised concerns about the speed with which MoLE produced diagrams. It should be noted that the software was, where possible, run on PowerPC Apple computers. Due to equipment constraints, subject 31 was using a version running on a Macintosh IIfx. Since no other subjects commented on this, the comment was taken to be a reflection on the hardware, rather than MoLE.

The fourth theme identified was error correction. It was clearly expected that the software would intelligently spot mistakes. In practice, MoLE's design relies on restricted methods of inputting information and automated processes (e.g. diagram construction) to reduce the number of mistakes, as described in section 4.2.

Related to this theme was the expectation of intelligent support from the software.

I'm assuming that there would be on-line help, and something to tell you when and why you are making errors in constructing the proofs and this would obviously make it easier to learn and also be more useful than referring to text and writing it all down on paper. I felt because I was constrained just to the text for help I was very unguided, because sometimes it was hard to follow the text, which made matters worse for me!

(Subject 21; pen & paper)

At present, MoLE is designed as a resource to complement teaching texts, whose role is to provide guidance and instruction; it is not designed to include such support, although the inclusion of such features and the development of MoLE into an intelligent learning environment could form the basis of future research work.

Subjects also expected the software to constrain and guide users. Although these comments have been grouped separately from comments on intelligent support (they imply passive guidance, rather than intelligent response), the conclusion remains the same: extending the support MoLE offers would involve changing the design, in this instance into a computer assisted learning package rather than a learning environment.

Two subjects raised more general concerns about the software. One found it hard to learn to use, although the subject unfortunately failed to provide specific examples of what they found difficult. The other was distracted by the software, experimenting with it to the detriment of concept development.

I spent the time playing with the software without grasping the specific difference between the three systems.

(Subject 38; MoLE)

A sixth theme was note taking. Although only one comment explicitly suggested that on-line support for this was desirable, others imply similar sentiments.

Finally, there was the expectation that the software would have motivational benefits:

Seems less like work, and you don't have to do all of the tedious stuff.

(Subject 14; pen & paper)

Such an expectation is unsurprising. The motivational benefits of MoLE have been demonstrated in section 6.4.

Summary

An analysis of the qualitative data revealed several findings which support or enhance the claims made in sections 6.3 and 6.4; i.e. that the grid world representation was easy to use, and that subjects show a preference for using this representation as part of a software package.

Concepts identified as being problematic in chapter three, such as Modal systems and proof construction, were explored further. Subjects' problems with Modal systems have been clarified: these stem primarily from difficulties understanding operator reduction, and from the hierarchical nature of the systems, where failure to understand one system prevents further progress. However, MoLE addressed the major concern raised about Modal systems, which was that subjects did not have enough time to learn about the different systems. MoLE also appears to have helped with the conceptual problems of proof construction.

Five further issues were identified and investigated. Of these, the most important was the importance of reducing the effort associated with routine tasks, supporting MoLE's design criteria. Also important was time pressure; again, this was less of a problem for MoLE users.

Comments about reflection and deeper learning remain indecisive, although there is quantitative support (such as ability to accurately assess understanding) to suggest that MoLE users had a deeper level of awareness of key concepts. Comparable evidence is lacking for the benefits of the pen and paper condition.

Problems with the level of abstraction also indicated a benefit to using MoLE, in spite of theoretical predictions that MoLE would be more abstract. It is suggested that this might point to the need to take into account users' perceptions when working with cognitive dimensions.

Finally, expectations of the software, potential refinements to MoLE, and a direction for future research work in this area (developing MoLE as an intelligent learning environment) were identified.

6.6 Discussion

Instructional condition

Instructional condition played a central role in this study, with subjects using MoLE performing better on a range of performance measures than subjects who had used pen and paper.

Qualitative data supported and developed these results, and confirmed that the blocks-world 'content' representation was effective for learners.

Three measures suggested that MoLE users had a wider grasp of the concept of Modal systems than their counterparts. The first was that MoLE users seem to be able to explain more of the Modal systems than subjects who had worked with pen and paper. The second was that subjects using MoLE were significantly better at applying Modal systems to operator reduction problems. Closely related to this was the trend for MoLE users to be better at dealing with abstract

operator reduction problems. This reflects an awareness that Modal systems operate independently of the examples they are applied to, and suggests a deep rather than just a procedural level of understanding.

MoLE allowed subjects to work faster than their counterparts, completing significantly more exercises during the intervention. However, this alone failed to account for the differences in scores. Re-scoring was carried out on the operator reduction problem (the only measure directly linked to how far through the material the subject had progressed), limiting marks to the material both groups had covered. This confirmed that MoLE allows subjects to work through more material *and* to learn more effectively.

Subjects' comments suggest that MoLE users also developed a better understanding of the process of proof construction. Although both groups were equally capable of constructing the proofs, MoLE users were better at assessing whether or not they had completed this process correctly. They also had far fewer conceptual problems than their pen and paper counterparts.

MoLE users were also more confident about their performance, and were more positive about the instructional condition they had been assigned to than subjects who had worked with pen and paper.

Some measures were unaffected by instructional condition. No differences were found between the instructional conditions for subjects' ability to explain accessibility relations, performance on the near transfer question, or for how much they had enjoyed the study.

Expertise

Expertise played a very limited role in this study. It was only a significant factor for one measure of improvement: how well subjects learnt to use examples or counter-examples during proof construction. It is not altogether surprising that expertise influenced this; such strategies are also used with other systems of logic, and will already be familiar to subjects who have studied formal systems. The only other measure which showed an effect for expertise was enjoyment, with those who had studied logic before enjoying the study more than those who had not.

The limited role of expertise was unexpected. It implies that using complementary representations is effective even for subjects with no prior experience of formal logic.

This contrasts with the study described in chapter three, where novice-expert differences were found. In that study, a mismatch between the representations used to teach and assess subjects may have accounted for this result. The mismatch caused particular problems for subjects in the learning environment condition; only experts in this instructional condition had previously used formal notation. To remove this problem in the current study, one formal symbol (implication) was used, and questions were presented in natural language as well as symbolically.

Given that subjects in this study were able to demonstrate an understanding of Modal concepts, irrespective of previous experience with formal logic, and given also that there was an effect of expertise in the previous study, it seems reasonable to conclude that the extent of subjects'

difficulties with Modal concepts may be hidden by problems relating to the use of formal notation. The comparatively strong performance of subjects in this study (e.g. 68% able to use Modal proofs, as opposed to 38% in the study described in chapter three), alongside the fact that 66% of respondents in the previous study had problems interpreting symbols (an issue not raised by subjects in this study), supports to this conclusion.

This points to a need for future research, extending tools such as MoLE in order to teach notational skills, as well as concepts. Programs such as Hyperproof (Barwise & Etchemendy, 1994), which extended the system created for Tarski's World, may provide a useful model for such developments. Alternatively, research could address the problem of what the domain-based tests are assessing. This would involve looking at whether assessment should change its focus from the manipulation of formal expressions to understanding the concepts involved.

There were two interesting trends for an interaction between experience and instructional condition. The first, with novice MoLE users learning how to use examples during proof construction, provides evidence for the design claim that the software's 'example' feature would support the acquisition of this strategy. The second, where expert MoLE users learned how to transfer knowledge to a new system, suggests simply that MoLE is more effective at promoting transfer than traditional forms of teaching. However, the fact that this was only a trend, and that no similar effect was found for novices, suggests that the tool is only providing weak support for the development of this process.

Gender

Whilst it seemed initially that gender had only one role to play in this study (an effect on enjoyment in participating in the study), it also had an unexpected effect of on the time-modified score for operator reduction. No obvious explanation of this was found.

Domain-specific problems

In addition to the quantitative findings, an analysis of the qualitative responses gathered in the post-test confirmed that the anticipated concerns (problems with the Modal systems and proof construction) were difficult for subjects, but that using MoLE helped address these. It also allowed these problems to be investigated further, revealing that the hierarchical nature of Modal systems and the meaning of operator reduction could become impediments to understanding.

Other factors influencing learning

Five further topics were also derived from the post-test comments. The first of these, reduction of effort, supports the criteria which shaped MoLE's design. The automation of routine or 'book-keeping' tasks was one of the common features identified in section 2.3 of software intended to support students learning logic. Subjects commented that having to work through routine tasks such as the construction of diagrams prevented them from concentrating on the concepts being taught.

Another commonly addressed issue was time pressure. Such comments were significantly more likely to come from subjects who had worked with pen and paper, reinforcing the finding that MoLE allows subjects to cover more material in the same amount of time. This, too, is in line with MoLE's design.

The third theme discussed concerns about reflection and deep learning. Interestingly, most of the comments on this topic were from experts. The comments were split evenly between arguing for the benefits of having the computer manage routine tasks, and for working with pen and paper in order to promote reflection. No quantitative evidence was found which supported the claim that using pen and paper promoted reflection; by contrast, several findings (such as ability to work with abstract examples, and to accurately assess understanding) could be argued to indicate deeper learning in MoLE users.

Abstract and theoretical concepts were another important cause for concern. Although the existence of this category was unsurprising, what was not anticipated was that 92% of comments were from subjects who had worked with pen and paper. This is made all the more interesting since, in terms of cognitive dimensions, the representation used in MoLE had a higher level of abstraction than that in the pen and paper condition. One suggested explanation for this, which matches the criteria used to design MoLE, is that the software offers a greater sense of 'concreteness', resulting from the ability to interact with and directly manipulate the blocks-worlds. This *sense* of concreteness seems to mitigate the effect of the level of abstraction. This explanation also provides support for the importance of the second common feature of software tools identified in section 2.3: the ability to offer examples which demonstrate and illustrate the concepts and skills being taught.

The mitigating effect of the users' perception raises an interesting question about cognitive dimensions. The dimensions are intended to describe 'information artifacts'; these may be paper-based, part of a software package, or even action-based (Green, 1994). Discussions are based on the perspective of "a single preferred cognitive strategy" or "a clear model of user activity" (Green, 1989). How does this account for the apparent mismatch between the raised level of abstraction of the implemented representation and the lack of comments from subjects who used it?

Reformulating cognitive dimensions to include the user's current understanding of the artifact may provide a way of resolving this problem. Such a model could be extended to cover the dynamic process of conceptual change which takes place whilst the user learns about the system. Additionally, the comparison of such a reformulated set of dimensions with one based on the 'ideal' model (for example, comparing the *perceived* level of abstraction with the theoretical one) may provide insight into how resilient to misinterpretation artifacts are, and how the user's current use of the artifact could be supported or improved. This is clearly an important potential area of research, which warrants further investigation.

The final theme drawn from the qualitative data included expectations and criticisms of MoLE. These criticisms suggest several possible refinements to the program, together with possible

future research which would involve redesigning MoLE as an intelligent learning environment. The other comments — features anticipated by subjects — are interesting both as an insight into the users' expectations, and as a framework for designing software tools. Specifically, these expectations included intelligent support and guidance, error checking, the automation of routine tasks, supporting clear presentation, and the use of visualisation and direct manipulation to promote understanding.

6.7 Conclusions

Chapter three demonstrated that using complementary representations to provide both support and content for formal reasoning gave effective support for students learning Modal logic. The software tool MoLE enhances this representation by supporting direct manipulation, constraining errors, offering examples to explain proofs, and automating rote procedures.

The study described in this chapter had three main aims: (i) to establish advantages of using MoLE; (ii) to investigate why certain topics in Modal logic are conceptually harder than others; and (iii) to identify further revisions which could usefully be made to MoLE.

The evaluation of MoLE confirmed that the software is more effective for learners than working with the same kind of representation manually. The tool allowed users to cover more material than their counterparts, and to learn the topics they covered more thoroughly. MoLE also appeared to promote a deeper understanding of key topics, with subjects showing a greater awareness of their skill at constructing proofs and an ability to distinguish the generic characteristics of Modal systems from the particular cases used as examples.

Additionally, MoLE had motivational benefits. 80% of subjects said they would have preferred to use MoLE than pen and paper. Although there was no difference between groups in terms of how much subjects had enjoyed the study, MoLE users were significantly more confident about having improved than subjects who had worked with pen and paper. Finally, MoLE users were more likely to have something positive to say about their experiences in the study.

Subjects who had used pen and paper to work through the instructional material performed no better than MoLE users on any measure of performance or motivation. This suggests that there is no pedagogic 'down-side' to using MoLE; it is at least as effective in this context (self-instruction) as *not* using the tool in all important respects.

Responses from 92.5% of subjects confirmed that blocks-world examples are easy to understand. This, together with the fact that both instructional conditions improved considerably from pre- to post-test, supports the finding from the study in chapter three that the combination of law-encoding diagrams with simple examples is an effective one.

The importance of using software tools to reduce the effort involved in carrying out routine tasks was stressed. Several subjects commented that such tasks hindered concept acquisition, a belief supported by the experimental findings and in line with cognitive load theory.

An analysis of comments addressed the study's aim of investigating why certain topics in Modal logic are conceptually harder than others. This showed that the hierarchical nature of Modal

systems and the meaning of operator reduction could become impediments to understanding more advanced topics.

Problems with abstract and theoretical concepts were also discussed. These comments revealed a strong imbalance between instructional conditions, with only 8% of the comments being from MoLE users. The explanations proposed for this discuss MoLE's emphasis on natural language rather than symbols, but also address a more general issue: that learning environments such as MoLE, which allow students to directly manipulate simple graphical examples, make the topic appear more concrete to learners. This leads to the proposal of a distinction between the representation's theoretical level of abstraction and the *perceived* degree of abstraction which users experience. Whether or not other cognitive dimensions are also dependent on users' understanding of information artifacts will need further investigation.

Finally, subjects' comments were used to identify a set of expectations about the features a software tool should incorporate. These focused on provision of intelligent support (error correction, help, guidance, etc), the automation of routine tasks, and the role of visualisation and manipulation in supporting understanding. They also identified useful revisions to MoLE which could form the basis of further research.

These findings provide answers to several of the research questions identified in section 1.3; for example, problems associated with learning Modal logic, or whether the benefits of using visualisation and manipulation tools are purely the result of the representation they incorporate. In the final chapter, these questions will be reviewed in order to assess the extent to which the research described in this thesis has provided answers to these problems.

Chapter7

Conclusions

7.1 Introduction

This chapter presents the conclusions of the research described in this thesis. It begins by working through the research questions identified in section 1.3, sifting the findings of the previous chapters in order to show the extent to which these have been answered. Contributions to knowledge are then identified, the limitations of the research are discussed, and the chapter concludes by identifying further related work which could be undertaken.

7.2 A review of the research questions

This thesis set out to investigate how visualisation and manipulation tools support students learning about formal reasoning. This section provides an answer to this question by reviewing the findings of the previous chapters.

In section 1.3, eight subsidiary questions were identified which, between them, provide an answer to the general research question. These fall into three broad categories: topic-specific issues (question 1), representational questions (questions 2-4) and software issues (questions 5-8). Each is considered in turn.

Topic-specific issues

What are the problems associated with learning Modal logic?

Seven key topics were identified which could pose conceptual difficulties for learners: modes, possible worlds, accessibility relations, Modal systems, Modal proofs, problems of reference, and issues of belief. The empirical study described in chapter three showed that these topics are not all equally difficult. Understanding necessity and possibility, using possible worlds, and issues of belief are concepts which were easily mastered by most of the students. Other areas, such as Modal systems, reference, and Modal proof construction proved to be much more difficult to learn.

Several concept-independent problems are also associated with learning logic. These include motivational concerns, problems relating formal methods to real-world situations, a lack of formal skills, and a difficulty in overcoming the gulf between natural and formal languages. In the study described in chapter three, two thirds of subjects found symbols hard to interpret, which is symptomatic of a combination of a lack of formal skills and the lack of familiarity with formal languages. They also commented on being able to apply skills without understanding what they meant — a clear indication of a surface level of understanding. The link between motivation and difficulty was also demonstrated, although this was shown to be more complex than was previously believed. Finally, few subjects in either of the studies described in chapters three and five were able to transfer skills to a related but unseen Modal system.

Representational questions

What uses of graphical representations does current research suggest will effectively support formal reasoning?

Existing research shows that graphical representations can be used effectively for a range of tasks, including the provision of overviews, supporting secondary notation, searches (where guidance can be provided by graphical constraints), and explicitly representing relationships. For reasoning tasks, specificity theory suggests that representations containing little or no abstraction will help to keep inferences simple. Several problems with using graphical representations were also identified, including issues of cognitive style and the need for users to learn how to use the representation.

Six schemes were considered which classify representations by type and by function, and that introduce a series of 'dimensions' along which representations can be compared. An analysis of these led to a simple but robust distinction between types of representation. This separates those well suited to providing content for reasoning from those that are most effective at supporting the process of reasoning. This distinction has, therefore, identified two uses of graphical representations which effectively support formal reasoning.

Applying this content/process distinction to representations of Modal logic highlighted the weaknesses of the representations traditionally used to teach this topic. It also led to the derivation of a new style of representation for Modal logic, referred to as the learning environment representation.

Are these predictions supported empirically?

The study presented in chapter three demonstrated that the instructional material based on the learning environment representation was more effective for learners than the syntactic or diagrammatic materials traditionally used to teach this topic. Qualitative data indicated that this difference was a result of the representations used, a finding supported by the analysis of the representations in terms of cognitive load and cognitive dimensions. Furthermore, a case study showed that use of an appropriate content representation (based on the blocks-world metaphor) can address problems such as a lack of prior formal skills, understanding possible worlds and interpreting the abstract, formal language of Modal logic.

Can existing representations, such as the blocks-world metaphor of Tarski's World, be adapted for use in other domains?

As the answer to the previous question shows, qualitative data from a case study demonstrated the effectiveness of the blocks-world representation in addressing conceptual problems in Modal logic. This demonstrates that this representation, which was created to support students learning first-order logic, can be successfully adapted for use in a related domain.

Additionally, the case study illustrated that this representation could be 'personalised', adapted and rebuilt by the student. Creating a new system of representation (based on the blocks-world) helped this student to understand several of the concepts that had previously eluded them. Comments from other subjects suggested that they, too, would have benefitted from building their own system in this way.

Software issues

What elements of existing programs are used to support formal reasoning?

The review of software identified two features commonly used to support students learning formal reasoning. The first was the automation of 'book-keeping' tasks, such as entering logical notation or proof construction. This was argued to promote learning by encouraging exploration of the domain. In chapter four, support for this argument was presented in terms of cognitive load theory. The second feature was the provision of examples. Using examples is argued to address motivational concerns and concept-independent problems such as the perceived lack of relevance of formal methods. Support for this is provided by the graphical representations literature used to justify 'content' representations.

What software design will incorporate these elements with the representation best suited to supporting students learning Modal logic?

Chapter four reported on the design of MoLE, a piece of software which incorporated complementary content and process representations into a learning environment for Modal logic. This was achieved by adopting the blocks-world 'content' representation as basis for the learning environment. The 'process' tree diagram was used as a tool to support the construction and interpretation of this environment.

The segmented structure of the software design (a series of pieces and grid worlds, connected together by the process representation) made it ideally suited to an object-oriented implementation. The Model-View-Controller formalism of Krasner & Pope (1988), adapted so that it supported a hierarchy of linked Model-View-Controller mechanisms, provided a structure capable of supporting this design.

What benefits do learners get from using this kind of tool?

MoLE incorporated several features which were found to be of benefit to learners. The study in chapter five showed that a visualisation and manipulation tool which incorporates complementary representations reduces the perceived abstraction of the topic. An interesting aside is that this finding runs counter to an analysis of the representations using cognitive

dimensions (section 6.6). Another of MoLE's features, the generation of illustrative examples, helped learners to develop a deeper understanding of the skills and concepts they acquired.

As with the programs reviewed in chapter two, the automation of rote tasks (such as constructing diagrams) was argued to support learning. Automation reduces cognitive load, constrains errors, and allows greater exploration of the domain by making examples easier and quicker to work through. The evaluation of MoLE in chapter five provided empirical support for these claims.

Are these benefits merely the result of the representation, or is there added value in incorporating the representation into a software tool in this way?

The evaluation in chapter five showed that working with software, rather than pen and paper, offered several benefits. These advantages are the result of incorporating the representation into a software tool.

Students who used MoLE learnt faster and more thoroughly, out-performing their counterparts even on material that both groups had covered. MoLE users also had a better appreciation of the skills Modal logic requires, which meant that they were more accurate at assessing how well they had performed. They were also better at abstracting general properties from particular examples. These results pointed to a deeper style of learning than students who worked with pen and paper.

As mentioned above, a further benefit of the software was that increasing the interactivity of the content representation by using them as microworlds, rather than static illustrations, reduced the perceived level of abstraction of the system.

Summary

This section has revisited the eight subsidiary research questions specified in chapter one, showing the extent to which each has been answered by the investigation described in this thesis. This has shown that not all concepts in Modal logic are equally difficult; some, such as the idea of Modal systems, are far more difficult than others. It has also illustrated the importance of the distinction between content and process representations, which has proved central to this research. Finally, it has shown that there is considerable added value for learners in using these complementary representations as the basis of a visualisation and manipulation tool that supports the automation of routine, 'book-keeping' tasks.

7.3 Contributions to knowledge

Several of the findings of this thesis can be seen as contributions to knowledge. This section describes how these fit into research in the fields of Modal logic, learning with graphical representations, software design and cognitive science.

Modal logic

The results of the study in chapter three have led to the development of a grounded theory (Glaser & Strauss, 1967) of learning Modal logic. This was complemented by the analysis of quantitative data on measures of conceptual performance in chapters three and six.

The findings of these chapters indicated that not all concepts are equally difficult.

Understanding and applying of Modal systems, together with the construction of Modal proofs, were the most difficult topics for students to master. By contrast, students found the concepts of modes, possible worlds and accessibility relations fairly simple. The concepts of belief and reference fell somewhere between these extremes.

Importantly, the findings of chapters three and six gave a clearer understanding of the role of formal skills in logic learning. In spite of performing as well as subjects in the other conditions on all measures of performance and motivation, subjects in the learning environment condition exhibited a novice-expert difference. This was explained by the emphasis on formal skills in the assessment, which was not present in their intervention material. The study in chapter six showed no expert-novice differences in performance for students in either condition; here, the assessment examined conceptual skills and understanding, rather than formal ones. These two findings suggest that prior formal training does not affect conceptual development, but instead influences students' ability to express this knowledge formally. The previous assumption that prior formal training influence logic learning, rather than formal skills, can be explained by the emphasis on the formal expression of knowledge predominant in research and teaching.

In addition, the following specific results were identified:

- Examples help to illustrate concepts, making them more meaningful for learners.
- Abstract notation is hard to interpret and can intimidate learners.
- Interpretation of formal notation can be eased by providing contextualised examples.
- 'Real world' examples of possible worlds can be too rich to makes sense of.
- Manipulation skills can be acquired without being understood.
- Negative propositions appeared to be harder for students to interpret than positive ones.
- Double negatives can cause students confusion.
- Low motivation and perceptions of difficulty form a vicious circle. This both supports and extends previous findings linking the difficulty of logic with motivation (e.g. Fung & O'Shea, 1994).
- Self-constructed examples and explanations help overcome conceptual problems and enhance motivation.
- Students can have difficulty interpreting strings of Modal operators.
- The order of evaluating Modal operators is ambiguous (should they be read left to right, or 'inside' to 'outside'?)
- There is a perceived qualitative difference, in terms of the problem of reference, between symbols and names, with the association between sense and reference seen as being closer for names.

- Students found system T comparatively easy to learn; systems S4 and S5 were seen as progressively harder.
- The progression from system T to S4 and on to S5 means that problems understanding one system will block the acquisition of more advanced topics.

The primary application of these findings is in the design of teaching materials for Modal logic. Understanding the obstacles to learning is clearly an important pedagogical step. These findings can also be used for learning resources such as intelligent tutoring systems, providing a simple model of the problems associated with learning Modal logic.

Chapter six confirmed and developed these findings. This also showed that this classification of problems can be used as a framework with which to evaluate students' ability in Modal logic.

Learning with graphical representations

Research in artificial intelligence concerned with the use of graphical representations has proposed a series of classification systems intended to support representation selection. A system which classifies representations by role, distinguishing between content and process representations, contributes to this research in two important ways.

Firstly, the distinction remains constant across existing classification systems. This suggests that the distinction can be used to provide common ground in a field which is currently riddled with individual systems and terminology.

Secondly, the two roles for representations are argued to be complementary. The studies in this thesis confirm that use of these complementary representation can provide effective support for students learning an abstract topic. In the study described in chapter three, students using the diagrammatic representation performed worse than those in other conditions on most measures of improvement. Although the performance of students working with the syntactic and those using the learning environment representations was comparable, two considerations were used to justify the conclusion that the learning environment representation was more effective. Firstly, it had motivational benefits, addressing one of the key learning difficulties identified in the literature review. Secondly, qualitative data highlighted a mismatch between the representations used by subjects in the learning environment condition for instruction and assessment. Because these students were unfamiliar with the syntactic notation used in the post-test, their performance (already equal to that of students in the syntactic condition) was impeded.

The difference in performance between conditions was explained by analysing the three representations using Green's cognitive dimensions. This showed that the diagrammatic representation required a considerable degree of premature commitment, involved a high level of abstraction, and was the most viscous of the three forms of representation.

Several further findings are also relevant to research into learning with graphical representations:

- Concrete representations can be used to support students learning abstract topics, enhancing motivation and helping them to understand the skills and concepts they acquire. Students found this type of representation easy to work with.
- Students using concrete representations commented that it made the topic seem “more real”, and “less abstract”.
- Concrete representations can be re-used in related domains, requiring only minor modification in order to preserve an appropriate level of expressivity.
- Comments suggest that non-pictorial graphical representations are not always considered to be ‘diagrammatic’; they may be viewed as abstract and even syntactic, even if they possess qualities (e.g. hierarchical ordering, grouping, etc.) that are essentially graphical.

These results contribute to the ongoing discussion on the use of multiple representations to supporting learning.

Software design

Chapter four describes the model used as a basis for the development of the software tool MoLE. This design incorporates three innovations:

- The use of complementary representations as the basis of a learning environment.
- An extension of the Model-View-Controller formalism described by Krasner & Pope (1988), which enabled it to support linked Model-View-Controller mechanisms. This provided a more sophisticated design able to support the use of complementary representations.
- The ability to offer examples and counter-examples for Modal proofs, helping to illustrate the concepts involved.

MoLE also aimed to reduce the cognitive load of users involved in problem solving by automating routine procedures such as drawing diagrams, and sought to make the content representation more effective by supporting direct manipulation.

The robustness of the design was demonstrated in chapter five, confirming that the extended Model-View-Controller formalism provides a useful basis for learning environments.

The empirical study described in chapter six concluded that automation of rote tasks is of great importance, as it allows learners to focus on concepts rather than procedures. The benefits of this can be seen in the greater accuracy of self-assessment and deeper understanding of concepts from learners who used MoLE.

This study also showed the importance of direct manipulation in abstract domains. Students perceived the topic as being less abstract than their counterparts, who used the same representation but worked with pen and paper rather than software.

The adaptation of elements of Tarski’s World illustrated that software designs based on content representations are as suitable for adaptation and re-use as the representations themselves.

Finally, the study with MoLE described in chapter six provided an opportunity to investigate students' expectations of the features a software tool to support logic learning should offer.

These included:

- Intelligent support, help and guidance
- Automated error-checking
- Automation of routine tasks
- Support for clarity of presentation (e.g. automatically neatening diagrams)
- The use of visualisation and direct manipulation to support understanding

Cognitive Science

As demonstrated in chapter three, cognitive dimensions form a useful tool for the analysis of a wide variety of information artefacts and notational systems (Green, 1994). These dimensions are applied to "a single preferred cognitive strategy" or "a clear model of user activity" (Green, 1989).

The comments on the perceived abstraction of representations of Modal logic in chapter six have demonstrated that this formulation of cognitive dimensions, whilst useful, does not account for the different understanding of an artefact individual users may have. Whilst this contribution is, at present, a small one, it offers the opportunity to carry out an important investigation into individual differences and the use of cognitive dimensions, as section 7.5 will describe.

Specific contributions

In addition to the general findings described above, a number of more tangible contributions to knowledge have been made. Chapter two contains an overview of current literature on learning formal reasoning and on reasoning with graphical representations. The work described in chapters three and six has field tested instructional material for Modal logic (Appendices 1 and 8). These studies generated large amounts of data, both on how students learn this topic and on the role of graphical representations in reasoning (Appendices 2, 3, 4, 9 and 10). Chapters four, five and six have led to the production of a tool to support learners of Modal logic, MoLE, whose source code is provided in Appendix 6. The interface design of this tool has been tried, tested and refined, providing a template for further visualisation and manipulation tools. Finally, MoLE has also provided an adapted version of Smalltalk's Model-View-Controller formalism, as described by Krasner & Pope (1988), which is capable of supporting hierarchical models and complementary representations, as discussed in chapter four. Any of these resources could form the basis of future analysis and research.

7.4 Limitations of the current research

The research described in this thesis has a number of limitations. These range from pragmatic compromises to concerns that fall beyond the scope of the thesis. This section presents these limitations and explains their impact.

Combining graphical representations

The content and process representations used in this thesis were selected from the systems identified in the literature review in chapter two. Two criteria were used: previous effectiveness for students learning logic, and the extent to which they satisfied the requirements of the six systems of comparing representations described in section 2.4. In this case, the tree diagram is widely used by learners, is law encoding, an UARS, abstract, etc., whilst the blocks-world representation is concrete, depicts one state, is a MARS, and so forth.

Whilst these criteria can ensure that the representations selected are complementary, it cannot guarantee that combining them will prove effective for learners, since the effects of using multiple representations are not well understood. Existing criteria, such as those of Ainsworth (1997), have been considered; the successful use of complementary representations in this thesis can be seen as providing support for the principle, “use representations to support different processes”, which she puts forwards as a guideline for effective multiple representation use. However, fleshing out these guidelines into clear criteria for combining representations will require further research.

Software implementation

MoLE incorporated the features identified by the review of software as being beneficial to learners, and a formative evaluation of the program enabled further additions and revisions to be made.

Several further features were omitted which would have made MoLE a more fully-developed tool, and would enable it to support a broader range of learners. These include support for logical operators, or for graphical conventions for abstraction in the style of Hyperproof.

However, whilst it would have been interesting to investigate how these features were used, their absence had little impact on the main topics of this thesis. Including conventions for abstraction would have added another theme to the investigation of complementary representation use. Not supporting formal logical operators proved to be a useful limitation, as it focused attention on the concepts from Modal logic being investigated, rather than on familiarity with first-order logic, as the eradication of effects of expertise in chapter six demonstrated. The study in chapter three showed that having to deal with such notation in addition to learning about Modal logic contributed to a novice/expert difference on performance measures.

Empirical studies

Whilst the studies reported in chapters five and six successfully investigated the hypotheses under investigation, the design of questions in the study described in chapter three meant that subjects’ understanding of accessibility relations was indistinguishable from their performance at proof construction. Furthermore, the decision was taken to assess all three conditions using the same representational style. This had been intended to ensure consistency, but had the unpredicted side-effect of causing a novice-expert split in performance for subjects using the learning environment representation. This was because subjects who had not completed a prior course in logic were

unfamiliar with the notation that was used. This indicated that using three versions of the test material, one for each style of representation, would have provided a better measure of performance.

Also, these studies assess conceptual understanding rather than the ability to create formal proofs. Proof construction remains the key indicator of ability in this domain as assessed in degree-level courses. Altering the evaluations to reflect this might give greater validity to the research in terms of the current course contexts, and would have permitted a comparison with students taking traditional Modal logic courses. However, it would have precluded the emphasis on acquiring concepts, which has been central to this research.

7.5 Further Research

The research described in this thesis has attempted to answer the questions presented in the introduction. In the course of this research, several other areas of work have been identified that merit deeper investigation.

Further investigation of topics in Modal logic

Further research could be undertaken to investigate the processes involved in learning each of the seven topics from Modal logic covered by this thesis; for example, how students learn to construct Modal proofs. This would contribute to the emerging grounded theory outlined in chapter three, and would help to provide a deeper understanding of the problems associated with learning this topic. Another possibility for extending this work would be to investigate advanced concepts from Modal logic.

Further investigation of complementary representations

The research in this thesis provides a good starting point for a discussion of the use of complementary representations in supporting learning. It has also shown how content representations can be adapted for use in a new domain (in this case, from first-order to Modal logic).

One useful extension of this work would be to see whether successful process representations can also be adapted. A student could reason correctly about an ill-formed or unsuitable content representation, although this is clearly not ideal. By contrast, an inaccurate process representation could not be law encoding, which could hinder performance. The 'robustness' of such representations is an important issue, since it will indicate how tolerant a representation is to misinterpretations such as students' misconceptions.

A related course of work would be to investigate whether students can successfully adapt or personalise process representation. Correctly forming new representations requires an understanding of the rules being encoded. Analysing the differences between students' representations may provide insights into their misconceptions.

The transfer of representations to a new domain

The pairing of complementary representations adopting in this thesis was shown to be effective for learners of Modal logic. The choice of these was based on domain-independent systems of

classification. This suggests that complementary representations will also be effective in other domains, particularly those of an abstract nature.

The investigation of a closely related area of reasoning, such as fuzzy logic, would permit a deeper understanding of the ways in which existing representations can be adapted to suit a new domain. A series of studies which paralleled those described in this thesis would provide evidence of the transferability of representations, allowing insight into how they can best be reused in a variety of domains.

Comparison with other pairs of representations

Comparisons of the learning environment system (blocks-worlds and tree diagrams) with other pairs of representations would allow an interesting development of the work in this thesis. These alternative pairs could involve altering only one of the representations used in the learning environment system, allowing greater control over the comparisons.

Such comparisons may identify more effective content or process representations, test and refine the frameworks for comparing representations outlined in chapter two, contribute to ongoing discussions of multiple representation use, and elicit commonalities which can be used to develop the general theory of complementary representations. This would also address the concern that the effects described in this thesis may be atypical, resulting from this particular pairing rather than reflecting a general advantage of complementary representations.

Further development of MoLE

Just as Hyperproof was produced by enhancing Tarski's World, a more advanced visualisation and manipulation tool could be derived from MoLE by introducing conventions for graphical abstraction. Adding objects such as 'barrels' and 'bags' (Barwise & Etchemendy, 1994; see section 2.3), for example, would allow the tool to support a wider range of problems, making it suitable for more advanced courses of Modal logic.

Another possibility would be to provide more detailed support for topics such as reference, perhaps by allowing pieces to 'evolve' at each step of the diagram. Again, this would provide a broader scope for the tool, giving it a wider applicability.

Perhaps more pressingly, however, MoLE could be extended to support logical operators and quantifiers, allowing it to be better integrated into existing Modal logic courses. This would enable it to support students' understanding of formal notation, as well as Modal concepts.

A final possibility is that MoLE could be re-developed as an intelligent learning environment, in order to meet the expectations of students identified in chapter six.

Adapting the software to other domains

As well as investigating whether the content and process representations could be adapted for use in other subjects, it would be interesting to see how MoLE could be altered to suit new domains.

Only minor modifications would be required in order to extend the current system to a domain such as fuzzy logic. This is, in part, a reflection on the similarity between the domains.

However, it also follows as a consequence of having adopted the Model-View-Controller paradigm for the software design, which allows each element to be ‘unplugged’ and replaced individually. This would make it easy to substitute a new ‘process’ model, for example, which left the blocks-worlds unchanged, but enabled MoLE to support a new domain.

Investigation of the effect of users’ perceptions on cognitive dimensions

An unexpected result in chapter five suggested that interaction with the grid worlds may have altered users’ perceptions about how abstract the system was.

Further investigation of the role of users’ perceptions would be prudent. Several key comparison systems (e.g. specificity, functional classification) rely on a ‘theoretically ideal’ perspective, which assumes that the representation is fully understood and will be used in a standard manner. Since learning how to use a representation is typically a part of the process of learning about a topic, understanding of how these mis-interpretations may affect a representation’s usefulness is of great importance.

A first step towards this investigation would be to model users’ understanding of a representation (or any other information artefact) as they learn how to use it. This process would provide a longitudinal model of conceptual development, in terms of cognitive dimensions. Such a model may provide important insights into novice-expert differences, for example, by explaining why systems which are effective for experts are inappropriate for learners. Importantly, the generality of cognitive dimensions suggests that this procedure would be applicable in almost any domain.

7.6 Summary

This chapter has presented a review of the questions addressed in the thesis, and identified the extent to which these have been answered. Limitations of the research have been discussed, as have opportunities for further work.

To investigate the role of visualisation in logic learning, literature on the use of representations in learning was reviewed, and an important distinction was drawn between two ways in which graphical representations can be used to support reasoning: to provide content, or to support the reasoning process. These roles were argued to be complementary, and the efficacy of using combinations of such representations to support learning has been demonstrated.

The studies described in chapters three and six have identified problems central to the process of learning Modal logic. These also illustrated the importance of domain-independent factors, such as motivation, provision of examples, and ability to interpret formal notation, for students studying logic.

Research into the use of software to support students learning logic led to the design and implementation of MoLE, a learning environment for Modal logic. This incorporates an innovative environment based on complementary representations, enhances the content representation through direct manipulation, provides examples to illustrate the concepts being

taught, and automates routine and 'book-keeping' tasks. These facilities are supported by an adaptation of the Model-View-Controller formalism described by Krasner & Pope (1988).

Overall, the investigation described in this thesis aimed to address the important problem of how to design an effective visualisation and manipulation tool for learners of Modal logic. The conclusion to be drawn from this work is that such a tool should incorporate complementary types of representation, thus providing content and support for formal reasoning, and should automate rote tasks in order to effectively support learning.

References

- Ainsworth, S. (1997) Designing and Evaluating Multi-Representational Learning Environments for Primary Mathematics. PhD Thesis, Technical Report no. 47, ESRC Centre for Research in Development, Instruction and Training, Department of Psychology, University of Nottingham.
- Ainsworth, S., Wood, D. & Bibby, P. (1996) Co-Ordinating Multiple Representations in Computer Based Learning Environments. Proceedings of the 1st European Conference on Artificial Intelligence in Education, pp. 336-342, Lisbon.
- Barwise, J. & Etchemendy, J. (1990) Visual Information and Valid Reasoning. In Visualisation in Mathematics, Zimmerman W. (Ed.). Mathematical Association of America, Washington, America.
- Barwise, J. & Etchemendy, J. (1992) The Language of First-Order Logic: including the program Tarski's World. Center for the Study of Language and Information Lecture Notes No. 34, Stanford University, America.
- Barwise, J. & Etchemendy, J. (1994) Hyperproof. Center for the Study of Language and Information Lecture Notes No. 42, Stanford University, America.
- Bornat, R. & Sufrin, B. (1996) Animating the Formal Proof at the Surface: The Jape Proof Calculator. Technical report, Department of Computer Science, Queen Mary & Westfield College, University of London.
- Boulton, D., & Hammersley, M. (1996) Analysis of Unstructured Data. In, Jupp, V., & Sapsford, R., (Eds), Data Collection and Analysis. Sage Publications, London.
- Campbell, K., Collis, K. & Watson, J. (1995) Visual Processing During Mathematical Problem Solving. Educational Studies in Mathematics, Vol. 28, pp. 177-194.

- Chadrasekaran, B., Narayanan, N. & Iwasaki, Y. (1993) Reasoning with Diagrammatic Representations: A Report on the Spring Symposium. *AI Magazine*, Summer issue.
- Chellas, B. (1980) *Modal Logic*. Cambridge University Press, England.
- Cheng, P., Holyoak, K., Nisbett, R. & Oliver, L. (1986) Pragmatic versus Syntactic Approaches to Training Deductive Reasoning. *Cognitive Psychology*, Vol. 18, pp. 293-328.
- Cheng, P. C-H (1996a) Functional Roles for the Cognitive Analysis of Diagrams in Problem Solving. In Cottrell, G. (Ed.), *Proceedings of the 18th Annual Conference of the Cognitive Science Society*, pp. 207-212, Hillsdale, New Jersey.
- Cheng, P. C-H (1996b) Learning Qualitative Relations in Physics with Law Encoding Diagrams. In Cottrell, G. (Ed.), *Proceedings of the 18th Annual Conference of the Cognitive Science Society*, pp. 512-517, Hillsdale, New Jersey.
- Cheng, P. C-H (1996c) Scientific Discovery with Law-Encoding Diagrams. *Creativity Research Journal*, Vol. 9, No. 2 & 3, pp. 145-162.
- Cheng, P. C-H (1996d) Law Encoding Diagrams for Instructional Systems. *Journal of Artificial Intelligence in Education*, Vol. 7, No. 1, pp. 33-74.
- Cox, R. & Brna, P. (1995) Supporting the Use of External Representations in Problem Solving: the Need for Flexible Learning Environments. *Journal of Artificial Intelligence in Education*, Vol. 6, No. 2, pp. 293-302.
- Cox, R., Stenning, K. & Oberlander, J. (1994) Graphical Effects in Learning Logic: Reasoning, Representation and Individual Differences. *Proceedings of the 16th annual meeting of the Cognitive Science Society*, Atlanta, Georgia, pp. 237-242.
- Cox, R., Stenning, K. & Oberlander, J. (1995) The Effect of Graphical and Sentential Logic Teaching on Spontaneous External Representation. *Cognitive Studies: Bulletin of the Japanese Cognitive Science Society*, Vol. 2, no. 4, pp. 56-75.
- Cronbach, L. & Snow, R. (1977) *Aptitudes and Instructional Methods*. Irvington, New York.
- Dawson, M. (1991) *A Generic Logic Environment*. PhD Thesis, Imperial College of Science, Technology and Medicine, University of London.
- Draper, S., Mohammed, R., Byard, M., Driver, R., Hartley, R., Mallen, C., Twigger, D., O'Malley, C., Hennessey, S., O'Shea, T., Scanlon, E., & Spensley, F. (1992) *Conceptual Change in Science: Final Report to ESRC*. CALRG Technical report No. 123, Open University.
- Friedman, A. (1993) Designing Graphics to Support Mental Models. In Spector, J. M., Polson, M. C. & Muraida, D. J., *Automating Instructional Design: Concepts and Issues*. Educational Technology Publications, New Jersey.
- Fung, P. (1988) *Automated Diagnosis of Prolog Control Flow Errors: a First Evaluation*. CITE Report No. 55, Open University.

- Fung, P. & O'Shea, T. (1992) Learning to Reason Formally about Programs: An Observational Study of Computer Science Students. CITE report No. 168, Open University.
- Fung, P. & O'Shea, T. (1994) Using Software Tools to Learn Formal Reasoning: A First Assessment. CITE report No. 168, Open University.
- Fung, P., O'Shea, T., Goldson, D., Reeves, S. & Bornat, R. (1993) Computer Science Students Perceptions of Learning Formal Reasoning Methods. *International Journal of Mathematics Education in Science and Technology*, Vol. 24, No. 5, pp. 749-759.
- Fung, P., O'Shea, T., Goldson, D., Reeves, S. & Bornat, R. (1996) Computer Tools to Teach Formal Reasoning. *Computers in Education*, Vol. 27, No. 1, pp. 59-69.
- Glaser, B. & Strauss, A. (1967) *The Discovery of Grounded Theory*. Aldine, New York.
- Goldson, D., Hopkins, M., & Reeves, S. (1994) MiraCalc: The Miranda Calculator, the UNIX Version. Technical report, Department of Computer Science, Queen Mary & Westfield College, University of London.
- Goldson, D., Reeves, S. & Bornat, R. (1996) A Review of Several Programs for the Teaching of Logic. *The Computer Journal*, Vol. 36, No. 4, pp. 373-386.
- Good, J. & Brna, P. (1995) Novice Difficulties with Recursion: Do Graphical Representations Hold the Solution? *Proceedings of the 1st European Conference on Artificial Intelligence in Education*, pp. 336-342, Lisbon.
- Green, T. R. G. & Blackwell, A. F. (1996) Thinking About Visual Programs. *Proceedings of the IEE Computing and Control division's Thinking with Diagrams colloquium*, Digest no. 96/010, pp. 8/1-8/6, London.
- Green, T. R. G. (1989) Cognitive Dimensions of Notations. In Sutcliffe A. & MacAulay, L. (Eds), "People and Computers V", Cambridge University Press.
- Green, T.R.G. (1994) The Cognitive Dimensions of Information Structures. *Technical Communication*, Third Quarter, pp. 544-548.
- Hayes, P. (1993) I Can't Quite See What You Mean. *Computational Intelligence*, Vol.9, No. 4, pp. 381-386.
- Hughes, G., & Cresswell, M. (1985) *A Companion to Modal logic*. Routledge, London.
- Hughes, G., & Cresswell, M. (1985) *An introduction to Modal logic*. Routledge, London.
- Larkin, J. & Simon, H (1987) Why a Diagram is (Sometimes) Worth Ten Thousand Words. *Cognitive Science*, Vol. 11, pp. 65-99.
- Lemmon, E. (1977) An Introduction to Modal Logic. Segerberg, K. (Ed.); Monologue No. 11, *American Philosophical Quarterly*. Blackwell, Oxford, England.
- Levesque, H. (1986) Making Believers out of Computers. *Artificial Intelligence*, Vol. 30, pp. 65-99.

- Lovegrove, G., & Hall, B. (1991) Where are the Girls Now? In Lovegrove, G., and Segal, B., (Eds) *Women into Computing, selected papers 1988-1990*. Springer-Verlag, New York.
- Manna, Z. & Waldinger, R. (1985) *The Logical Basis for Computer Programming, Vol. 1: Deductive Reasoning*. Addison-Wesley, Reading, England.
- Oberlander, J., Cox, R., & Stenning, K. (1994) *Proof Styles and Multi-Modal Reasoning*. Proceedings of the 4th conference on Situation Theory and its Applications — Information Oriented Approaches to Language, Logic and Computation (STA94), Moraga, California.
- Oliver, M. (1995) How Situated Learning can Make a Meaningful Contribution to Modal Logic: Design Issues for a Software Tool. Proceedings of the AACE International Conference on Computers in Education, Singapore.
- Oliver, M. & O'Shea, T. (1996) Using the Model-View-Controller Mechanism to Combine Representations of Possible Worlds for Learners of Modal Logic. Proceedings of the first European Conference on Artificial Intelligence in Education, Lisbon.
- Oliver, M., O'Shea, T. & Fung, P. (1997) Three Representations of Modal Logic: A Comparative Experiment. *Machine Graphics and Vision*, Vol. 6, No. 1, pp. 109-128.
- Petre, M. (1993) Using Graphical Representations Requires Skill, and Graphical Readership is an *Acquired Skill*. Proceedings of the AI-Ed Workshop on Graphical Representations, Reasoning and Communication, Edinburgh, pp. 37-42.
- Petre, M. & Green, T.R.G. (1990) Is Graphical Notation Really Superior to Text, or Just Different? Some Claims by Logic Designers about Graphics in Notation. CITE report No. 113, Open University.
- Petre, M. & Green, T.R.G. (1993) Learning to Read Graphics: Some Evidence that 'Seeing' an Information Display is an Acquired Skill. *Journal of Visual Languages and Computing*, Vol. 4, pp. 55-70.
- Reeves, S., & Clarke, M. (1990) *Logic for Computer Science*. Addison-Wesley, Wokingham, England.
- Scaife, M. & Rogers, Y. (1995) External Cognition: How do Graphical Representations Work? *International Journal of Human-Computer Studies*, Vol. 45, pp. 185-213.
- Scaife, M. & Rogers, Y. (1996) External Cognition, Interactivity and Graphical Representations. Proceedings of the IEE Computing and Control division's Thinking with Diagrams colloquium, Digest no. 96/010, pp. 5/1-5/4, London.
- Shum, S. (1991) Cognitive Dimensions of Design Rationale. In Diaper, D. & Hammond, N., *People and Computers VI*. Cambridge University Press.
- Sloman, A. (1971) Interactions Between Philosophy and Artificial Intelligence: The Role of Intuition and Non-Logical Reasoning in Intelligence. *Artificial Intelligence*, Vol. 2, pp. 209-225.

- Stenning, K., Cox, R. & Oberlander, J. (1995) Contrasting the Cognitive Effects of Graphical and Sentential Logic Teaching: Reasoning, Representation and Individual Differences. *Language and Cognitive Processes*, Vol. 10, No. 3, pp. 333-354.
- Stenning, K. & Oberlander, J. (1995) A Cognitive Theory of Graphical and Linguistic Reasoning: Logic and Implementation. *Cognitive Science*, Vol. 19, No. 1, pp. 97-140.
- Sweller, J. (1988) Cognitive Load During Problem Solving: Effects on Learning. *Cognitive Science*, Vol. 12, pp. 257-285.
- Sweller, J. (1989) Cognitive Technology: Some Procedures for Facilitating Learning and Problem Solving in Mathematics and Science. *Journal of Educational Psychology*, Vol. 81, No. 4, pp. 457-466.
- Tennant, N. (1986) The Withering Away of Formal Semantics? *Mind and Language*, Vol. 1 No. 4, pp. 302-318.
- van der Pal, J. (1995) The Balance of Situated Action and Formal Instruction for Learning Conditional Reasoning. PhD Thesis, University of Twente.
- van der Pal, J. (1996) The Role of Interactive Graphics in Learning to Reason Abstractly. *Proceedings of the IEE Computing and Control division's Thinking with Diagrams colloquium*, Digest no. 96/010, pp. 14/1-14/3, London.

Appendices

Appendix 1: Study material, first study

Glossary of Symbols

- \rightarrow Implication, "implies"
- \sim Negation, "not"
- $\&$ Conjunction, "and"
- \vee Disjunction, "or"
- \leftrightarrow The biconditional, "if and only if"
(i.e. $(A \leftrightarrow B) \equiv_{\text{def}} (A \rightarrow B) \& (B \rightarrow A)$)
- \forall The universal quantifier, "for all"
- \exists The existential quantifier, "there exists"
- \diamond The modal operator possibility, "it could be the case"
- \Box The modal operator necessity, "it must be the case"

Truth Tables for First Order Logic

\rightarrow	T	F
T	T	F
F	T	T

$\&$	T	F
T	T	F
F	F	F

	\sim
T	F
F	T

\vee	T	F
T	T	T
F	T	F

\leftrightarrow	T	F
T	T	F
F	F	T

Please see overleaf for a summary of some key rules from first order logic.

Summary of Rules from First-Order Logic

1. Rule of Assumption (A)
Any proposition may be introduced at any stage as an assumption.
2. Modus Ponendo Ponens (MPP)
Given A , $A \rightarrow B$, we can conclude B .
3. Modus Tollendo Tollens (MTT)
Given $\neg B$, and $A \rightarrow B$, we may derive $\neg A$.
4. Double Negation (DN)
Given A , we can derive $\neg\neg A$.
5. Conditional Proof (CP)
Given a proof B from an assumption A , we can derive $A \rightarrow B$.
6. $\&$ -Introduction
Given A and B , we can derive $A \& B$.
7. $\&$ -Elimination
Given $A \& B$, we can derive either A or B separately.
8. \vee -Introduction
Given A , we can derive $A \vee B$.
9. \vee -Elimination
Given $A \vee B$, and the two proofs of C from assumption A , and C from assumption B , we can derive C .
10. Reductio ad Absurdum (RAA)
Given a proof of $B \& \neg B$ from an assumption A , we can derive $\neg A$.
11. \forall -Introduction
If $Fa \rightarrow Ga$ for an arbitrary object a , we can derive $(\forall x)(Fx \rightarrow Gx)$.
12. \forall -Elimination
Given $(\forall x)(Fx \rightarrow Gx)$, we can derive $Fa \rightarrow Ga$.
13. \exists -Introduction
Given Fa , we can derive $(\exists x)(Fx)$.
14. \exists -Elimination
Given $(\exists x)(Fx)$, we can derive Fa .

Pre-Test: Section A

Name: _____ Date: _____

Male ☐ Female: ☐

How would you rate your familiarity with logic?

- None ☐
- Some familiarity with the ideas..... ☐
- Familiar (e.g. know what all the symbols mean)..... ☐
- Competent (e.g. have completed a first course in logic)..... ☐
- Extensive (e.g. have also studied deviant logics) ☐

How long is it since you last studied logic?

- Have never studied it..... ☐
- Within a year ☐
- Between one and five years..... ☐
- Between five and ten years..... ☐
- Over ten years ☐

What course was it that had the logic content? (e.g. Prolog, philosophy, introduction to logic, etc.)

.....

How long is it since you last used logic?

- Have never made use of it..... ☐
- Within a year ☐
- Between one and five years ☐
- Between five and ten years ☐
- Over ten years ☐

Where did you make use of logic?

.....

Have you ever studied any of the following:

- Modal logic..... ☐
- Fuzzy logic ☐
- Three valued logic ☐
- Other non-standard logic..... ☐

(Please say which:)

Section B

Question 1

For each of the following sentences, say whether they could happen or not. Briefly explain why.

- a) Mount Snowdon is higher than Mount Everest.
- b) A yard is shorter than a metre, a metre is shorter than a chain, and a chain is shorter than a yard.
- c) I believe that 64 is the biggest possible number.
- d) $2+2 = 5$

Question 2

Say whether each of these sentences is contingent or not (i.e. that their truth values can change). Briefly explain why.

- a) All uncles are males.
- b) All males are uncles.
- c) All Blackbirds are black.
- d) All black birds are black.

Additionally, say which of these four are necessarily false, possibly true, or necessarily true.

Question 3

Say when, if ever, each of the following are valid. You may use the names T, S4 and S5 to summarise these conditions if you wish. 'p' denotes any proposition.

You may use either diagrams or symbols in your solutions.

- a) $\Diamond p \rightarrow \Box \Diamond p$
- b) $\Box p \rightarrow \Diamond p$
- c) $\Box \Diamond \Box \Diamond p \rightarrow \Box \Diamond p$
- d) $\Box(p \ \& \ q) \rightarrow (\Box p \ \& \ \Box q)$

Question 4

Let Q be a Modal system with the following property:

Q is not reflexive; in other words, worlds in Q are not related to themselves. This could also be expressed by saying that to calculate the truth value of a modal statement (such as "it is possibly true that..."), Q does not consider the values of statements in the initial world.

For this system, show that neither $\Box p \rightarrow p$ nor $p \rightarrow \Diamond p$

is valid.

Give an everyday example of when such a system could occur.

Intervention material 1

Time Started:

What is Modal Logic?

Modal logic is an extension of first-order logic. First-order logic deals with two concepts: the validity of a conclusion, and the truth of sentences. Whilst related, these are distinct qualities. If you assume that the sky is covered in material, and that all material is paisley, it's logically valid to conclude that the sky is paisley. Obviously, this isn't true.

However, sentences such as "The sky is blue" can change their truth value (it might be cloudy and grey, for example). First-Order logic cannot cope with changes; it merely describes what is actually the case at some given point in time. Nor can it cope with alternative situations ("What if...?"). The sky may be blue today, but did it have to turn out this way? These types of problems are called **contingencies**. Contingent sentences can be either true or false, depending on the situations they are evaluated in.

Examples 1: Which of the following are contingent?

- (1) Micky Mouse is older than Michael Jackson.
- (2) I am younger than the oldest person in the world.
- (3) $2 \times 3 = 6$
- (4) All black birds are black.

For a sentence to be contingent, we need to come up with at least one possible situation in which it can be true, and one in which it can be false. We can imagine a world in which Mickey Mouse was created before Michael Jackson was born, and equally, a world in which Mickey Mouse was created after Michael Jackson was born. Similarly, we can imagine a world in which I am younger than the oldest person in the world, and one in which I am not — if, for example, I was the oldest person in the world. However, $2 \times 3 = 6$ no matter what happens in the world — unless, that is, the symbols themselves change their meanings. Similarly, all black birds have to be black, because if they weren't, we'd have a logical contradiction. So, examples (1) and (2) are contingent, and (3) and (4) are not.

Possibility and Necessity

First order logic cannot deal with contingent sentences, and so Modal logic was created. This introduces the notion of **Possible Worlds**. Possible worlds are really nothing more than "What if...?" descriptions. To give an example of this, you could ask, "what if the sky were a different colour?". This question would result in a set of possible worlds, of which one possible world would be exactly the same as this one, but with a grey sky; another would be identical except for having a clear sky; another might have a grey sky and be snowing; and so on.

The problem with possible worlds is that there might be an infinite number of them, and it may not be at all clear how they relate to each other. Modal logic

imposes a structure on this set of worlds through **accessibility relations**. An accessibility relationship simply provides a way of linking one world to another. So, "wait five minutes", or "change one property of an object", "remove one object from this world", or even "do nothing" would all be valid accessibility relations.

Once possible worlds and accessibility relations have been defined, Modal logic sets about specifying what it means for something to be possible. A statement is **possible** (formally written as " \Diamond ") if it is true in at least one accessible possible world. From this definition, the notion of "necessity" can then be derived. A statement is **necessary** if it is not possible for it to be false ($\neg \Diamond \neg p$) — in other words, that it can't not happen. What this rather confusing definition actually means is that a statement is **necessary** (formally written as " \Box ") if it is true in every accessible possible world. Once these two terms have been defined, Modal logic can then deal with the sentences first order logic could not, since contingent statements are possible, and non-contingent statements are either necessarily true or necessarily false.

Examples 2: Are the following statements possibly true, necessarily true, or necessarily false?

- (1) Micky Mouse is older than Michael Jackson.
- (2) I am younger than the oldest person in the world.
- (3) $2 \times 3 = 5$
- (4) All black birds are black.

Looking at exercise 2, we can say that (1) and (2) are possible truths, since they are contingent. (3) is necessarily false, since there are no situations when it could possibly be true, and (4) is necessarily true, since (as before) there are no situations when it could possibly be false.

Modal Systems

The choice of accessibility relation is extremely important, since each works in its own way. For example, the relationship "change one property of an object" allows you to return to your initial possible world (you just change the property back), but "remove one object" doesn't. As you progress, you can see that the "remove one object" relationship will eventually reach a world in which nothing exists, no matter what route you take to get there. There is nothing comparable to this terminal state for the relationship "change one property of an object". This means that from the same starting point (i.e. the same original possible world), a statement which is possible with one accessibility relationship might be impossible (necessarily false) with another, and must be the case (necessarily true) with a third relationship.

Fortunately, many accessibility relationships behave in similar ways. This enables us to classify them into systems of Modal logic according to their properties. These properties will often be things like, "you can return to a previous state", or, "you can progress as many steps along this chain of states as you like". These tell us under what conditions conclusions will be valid. If, for example, you had two accessibility relationships called A and B, "p is necessary" might be A-valid but not B-valid. This means that relationship A will have

different properties to relationship B, and that, as a result, they will belong to different Modal systems. A simple way of summarising what properties need to hold for a conclusion to be valid is to say which system it will be valid in — so, a conclusion to a proof might be that “p is necessary” is A-valid.

The three most common systems are called T, S4 and S5. These will be defined below. For these definitions, note that W, W', W" and so on will denote possible worlds, and that R will denote an accessibility relationship. $W R W'$ means that W' is a possible world for W, using accessibility relationship R. This can be summarised by saying that W' is accessible from W.

One important thing about all three of these systems is that the truth values at the starting point, the initial possible world, needs to be considered too. This means that for most accessibility relationships, we will need to add “... or do nothing” in order to make the system work properly. Hence for all these systems, the relationship R is reflexive — i.e., $W R W$.

The system T describes perhaps the most basic types of relationships between possible worlds. The only requirement on this system is that it is reflexive. Importantly, this system allows us to “stack” Modal operators, such as $\Box\Box p$, or $\Box\Diamond p$, and so on. However, these operators can't be cancelled out at all. This makes T a very useful general system: by imposing no judgements on whether or not $\Diamond\Diamond p$ means something different from $\Diamond p$, it guarantees that any solutions it reaches will be accepted by everybody. However, the flip side of this is that fewer meaningful problems can be tackled, due to the complexity introduced by not canceling operators.

System S4 adds a second restriction to the relationships it allows. In addition to being reflexive, the relationship needs to be transitive. In other words, for worlds W, W' and W",

$$((W R W') \ \& \ (W' R W'')) \rightarrow (W R W'')$$

As noted with system T, it is possible to stack Modal operators. However, an alternative way to define S4 is by axioms, and it turns out that S4 is defined as having all the axioms of T plus one more: $\Box p \rightarrow \Box\Box p$. We already know that the definition of necessity can be derived from that of possibility, and so substituting in the formula $\Box p \stackrel{\text{def}}{=} \neg\Diamond\neg p$ we get:

$$\begin{aligned} \Box p &\rightarrow \Box\Box p \\ \neg\Diamond\neg p &\rightarrow \neg\Diamond\neg(\neg\Diamond\neg p) \\ \neg\Diamond\neg p &\rightarrow \neg\Diamond\Diamond\neg p \\ \Diamond\Diamond\neg p &\rightarrow \Diamond\neg p \end{aligned}$$

The last line is achieved by transposition, as in the first order logic strategy MTT. If we now define a new sentence, q, so that $q \stackrel{\text{def}}{=} \neg p$, we see that

$$\Diamond\Diamond q \rightarrow \Diamond q$$

It is even easier to prove that $\Box\Box p \rightarrow \Box p$, since for $\Box\Box p$ to be true, $\Box p$ must already be true in that world (from the definition of necessity).

What this means is that some Modal operators can be cancelled in S4. In fact, *any* repeated pattern can be removed.

The system S5 restrains relations still further. In addition to being reflexive and transitive, S5 is also symmetric, so that

$$(W R W') \rightarrow (W' R W)$$

An alternative way of specifying S5 is to take system T, and add the axiom,

$$\Diamond p \rightarrow \Box \Diamond p$$

The consequence of this is to allow even greater cancelling of Modal operators. This can be shown by a series of proofs which demonstrate that the rules for S4 also apply in S5; in other words, S5 cancels in the same way that S4 does, to repeated patterns. In addition, we can apply the definition of necessity to the new axiom and show that

$$\Box \Diamond p \rightarrow \Diamond p$$

Combining this result means that:

$$\Box \Diamond p \leftrightarrow \Diamond p \text{ (They are equivalent)}$$

In other words, we can cancel pattern of necessity of possibility down to possibility. The effect of this is that in S5, any string of Modal operators simply collapses to the last listed operator.

Examples 3: Operators in different modal systems

Formulae	In system T	In system S4	In system S5
$\Box \Box p$	$\Box \Box p$	$\Box p$	$\Box p$
$\Box \Diamond \Box p$	$\Box \Diamond \Box p$	$\Box \Diamond \Box p$	$\Box p$
$\Box \Diamond \Box \Diamond p$	$\Box \Diamond \Box \Diamond p$	$\Box \Diamond p$	$\Diamond p$
$\Diamond \Box \Box \Box \Diamond \Diamond \Box \Box \Box p$	$\Diamond \Box \Box \Box \Diamond \Diamond \Box \Box \Box p$	$\Diamond \Box p$	$\Box p$

Time Finished:

Intervention material 2

Time Started:

What is Modal Logic?

Modal logic is an extension of first-order logic. First-order logic deals with two concepts: the validity of a conclusion, and the truth of sentences. Whilst related, these are distinct qualities. If you assume that the sky is covered in material, and that all material is paisley, it's logically valid to conclude that the sky is paisley. Obviously, this isn't true.

However, sentences such as "The sky is blue" can change their truth value (it might be cloudy and grey, for example). First-Order logic cannot cope with changes; it merely describes what is actually the case at some given point in time. Nor can it cope with alternative situations ("What if...?"). The sky may be blue today, but did it have to turn out this way? These types of problems are called **contingencies**. Contingent sentences can be either true or false, depending on the situations they are evaluated in.

Examples 1: Which of the following are contingent?

- (1) Micky Mouse is older than Michael Jackson.
- (2) I am younger than the oldest person in the world.
- (3) $2 \times 3 = 6$
- (4) All black birds are black.

For a sentence to be contingent, we need to come up with at least one possible situation in which it can be true, and one in which it can be false. We can imagine a world in which Mickey Mouse was created before Michael Jackson was born, and equally, a world in which Mickey Mouse was created after Michael Jackson was born. Similarly, we can imagine a world in which I am younger than the oldest person in the world, and one in which I am not — if, for example, I *was* the oldest person in the world. However, $2 \times 3 = 6$ no matter what happens in the world — unless, that is, the symbols themselves change their meanings. Similarly, all black birds have to be black, because if they weren't, we'd have a logical contradiction. So, examples (1) and (2) are contingent, and (3) and (4) are not.

Possibility and Necessity

First order logic cannot deal with contingent sentences, and so Modal logic was created. This introduces the notion of **Possible Worlds**. Possible worlds are really nothing more than "What if...?" descriptions. To give an example of this, you could ask, "what if the sky were a different colour?". This question would result in a set of possible worlds, of which one possible world would be exactly the same as this one, but with a grey sky; another would be identical except for having a clear sky; another might have a grey sky and be snowing; and so on.

The problem with possible worlds is that there might be an infinite number of them, and it may not be at all clear how they relate to each other. Modal logic

imposes a structure on this set of worlds through **accessibility relations**. An accessibility relationship simply provides a way of linking one world to another. So, "wait five minutes", or "change one property of an object", "remove one object from this world", or even "do nothing" would all be valid accessibility relations.

Once possible worlds and accessibility relations have been defined, Modal logic sets about specifying what it means for something to be possible. A statement is **possible** (formally written as " \Diamond ") if it is true in at least one accessible possible world. From this definition, the notion of "necessity" can then be derived. A statement is **necessary** if it is not possible for it to be false ($\neg \Diamond \neg p$) — in other words, that it can't not happen. What this rather confusing definition actually means is that a statement is **necessary** (formally written as " \Box ") if it is true in every accessible possible world. Once these two terms have been defined, Modal logic can then deal with the sentences first order logic could not, since contingent statements are possible, and non-contingent statements are either necessarily true or necessarily false.

Examples 2: Are the following statements possibly true, necessarily true, or necessarily false?

- (1) Micky Mouse is older than Michael Jackson.
- (2) I am younger than the oldest person in the world.
- (3) $2 \times 3 = 5$
- (4) All black birds are black.

Looking at exercise 2, we can say that (1) and (2) are possible truths, since they are contingent. (3) is necessarily false, since there are no situations when it could possibly be true, and (4) is necessarily true, since (as before) there are no situations when it could possibly be false.

Modal Systems

The choice of accessibility relation is extremely important, since each works in its own way. For example, the relationship "change one property of an object" allows you to return to your initial possible world (you just change the property back), but "remove one object" doesn't. As you progress, you can see that the "remove one object" relationship will eventually reach a world in which nothing exists, no matter what route you take to get there. There is nothing comparable to this terminal state for the relationship "change one property of an object". This means that from the same starting point (i.e. the same original possible world), a statement which is possible with one accessibility relationship might be impossible (necessarily false) with another, and must be the case (necessarily true) with a third relationship.

Fortunately, many accessibility relationships behave in similar ways. This enables us to classify them into systems of Modal logic according to their properties. These properties will often be things like, "you can return to a previous state", or, "you can progress as many steps along this chain of states as you like". These tell us under what conditions conclusions will be valid. If, for example, you had two accessibility relationships called A and B, "p is necessary" might be A-valid but not B-valid. This means that relationship A will have

different properties to relationship B, and that, as a result, they will belong to different Modal systems. A simple way of summarising what properties need to hold for a conclusion to be valid is to say which system it will be valid in — so, a conclusion to a proof might be that “p is necessary” is A-valid.

The three most common systems are called T, S4 and S5. These will be defined below.

In order to represent these systems graphically, we will make the following conventions:

A possible world will be represented by a box, and each box will contain its name. Statements which are true in that world will be listed in distinct boxes, tacked onto the side. An example of this might look like figure 1, which is the first step of a proof investigating the truth of the sentence, $\Box(p \vee \Diamond q) \rightarrow (\Box p \vee \Diamond q)$.

W1	$\Box(p \vee \Diamond q) \rightarrow (\Box p \vee \Diamond q)$
----	--

Figure 1: an example of a possible world

From this, we can then work out the truth values. If we assume the statement “ $\Box(p \vee \Diamond q) \rightarrow (\Box p \vee \Diamond q)$ ” is false (denoted by a ‘0’ under the connecting arrow — see figure 2), then it follows that the first part must be true and the second false, since this is the only way that implication can fail. To mark section one is true, we put a ‘1’ under the \Box , and to mark section two false, we note ‘0’ under the connecting \vee , as shown in figure 2.

W1	$\Box(p \vee \Diamond q) \rightarrow (\Box p \vee \Diamond q)$
	1 0 0

Figure 2: A possible world the first truth values added

If $(\Box p \vee \Diamond q)$ is false, then both $\Box p$ and $\Diamond q$ must also be false, so we put a ‘0’ under each of those, too (figure 3).

W1	$\Box(p \vee \Diamond q) \rightarrow (\Box p \vee \Diamond q)$
	1 0 0 0 0

Figure 3: The updated possible world

All that remains is to work out the truth values of p and $\Diamond q$ in part one. But we know $\Diamond q$ to be false from the right hand side of the implication, meaning that p must be true. Additionally, for $\Diamond q$ to be false, q must be false too. This leaves us with the revised diagram, figure 4.

W1	$\Box(p \vee \Diamond q) \rightarrow (\Box p \vee \Diamond q)$
	$\begin{array}{ccccccc} 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array}$

Figure 4: the possible world with all truth values calculated

Having worked out the truth values of each element of W1, the next step is to look at the modal statements. First, to show that $\Box p$ is false, we need to consider a world in which p is false.

We can't use W1, since p is true. This means that we need to use a second possible world. The same might have to be done if we had a "possibility" statement that was true. To show this, an asterisk is placed under the false necessity terms and the true possibility terms. Each asterisk may need its own possible world.

Additionally, statements which must be true in all possible worlds (true necessity statements, and false possibility ones) are marked with an asterisk over them, as a reminder that what they assert must be true in each derived possible world.

This leads us to figure 5,

W1	$\overset{*}{\Box}(p \vee \Diamond q) \rightarrow (\Box p \vee \overset{*}{\Diamond} q)$
	$\begin{array}{ccccccc} 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array}$
	*

Figure 5: W1, showing which elements need a second possible world

Since the false expression $\Box p$ requires a possible world, we mark this with a line, indicating an accessibility relation. The world that is then constructed leads off to a third world in much the same way, resulting in the following final diagram (figure 6), whose consistency acts as a proof for the statement in world 1.

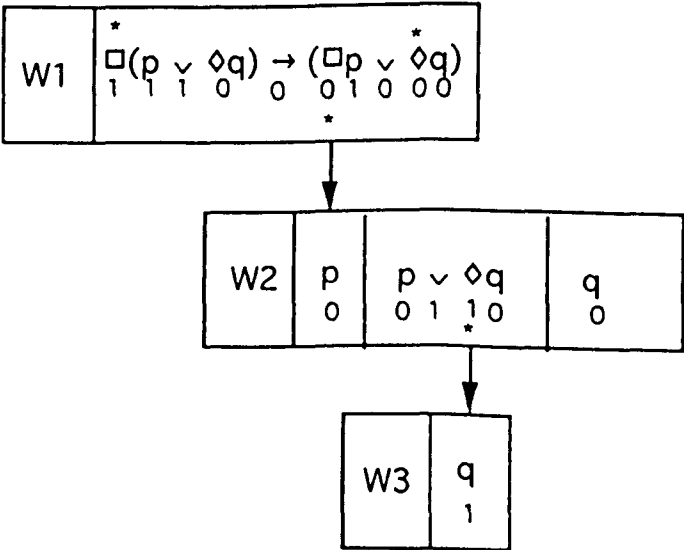


Figure 6: The final diagram

Another important attribute of S4 is that it allows worlds to refer “back up” the chain, resulting in loops of worlds, as shown in figure 8.

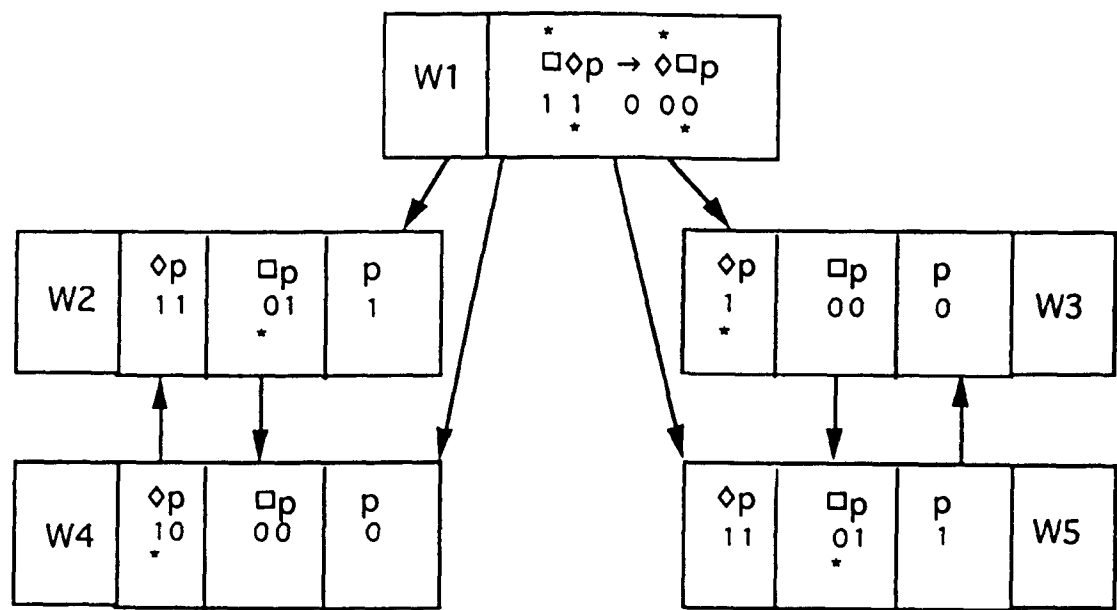


Figure 8: a diagram showing loops in a modal proof, e.g. with W4 and W2.
This diagram is consistent, showing it to be S4 valid

One of the interesting consequences of these loops is that when modal operators are stacked, *any* repeated pattern can be removed. This would mean that our

example of a stacked Modal operator for T, $\Diamond\Box\Box\Diamond p$, would contract to being simply $\Diamond p$. Something more elaborate, such as $\Diamond\Box\Box\Box\Diamond\Box\Box\Box\Box p$, would simplify to $\Diamond\Box p$.

The system S5 restrains relations still further. Here, each world has to be connected to every other world, as in the modified example in figure 9.

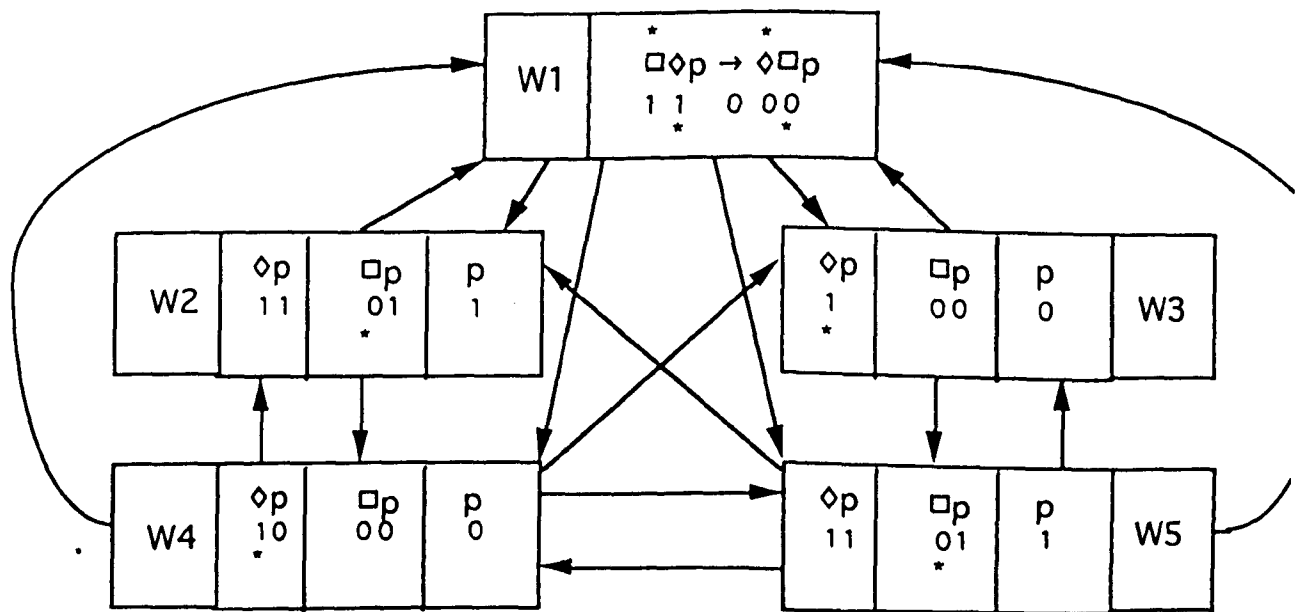


Figure 9: a diagram for S5 (which is consistent, showing it to be S5 valid)

This system has even more extreme effects on the way stacked Modal operators are tackled. It turns out that for this system, all Modal operators but the last can be ignored. Taking our earlier examples, this means that $\Diamond\Box\Box\Diamond p$ would collapse to $\Diamond p$, and $\Diamond\Box\Box\Box\Diamond\Box\Box\Box p$ to $\Box p$.

A table of examples of the different system's effects on stacked Modal operators is given below.

Examples 3: Operators in different modal systems

Formulae	In system T	In system S4	In system S5
$\Box\Box p$	$\Box\Box p$	$\Box p$	$\Box p$
$\Box\Diamond\Box p$	$\Box\Diamond\Box p$	$\Box\Diamond\Box p$	$\Box p$
$\Box\Diamond\Box\Diamond p$	$\Box\Diamond\Box\Diamond p$	$\Box\Diamond p$	$\Diamond p$
$\Diamond\Box\Box\Box\Diamond\Box\Box\Box p$	$\Diamond\Box\Box\Box\Diamond\Box\Box\Box p$	$\Diamond\Box p$	$\Box p$

Time Finished:

Time Started:

What is Modal Logic?

Modal logic is an extension of first-order logic. First-order logic deals with two concepts: the validity of a conclusion, and the truth of sentences. Whilst related, these are distinct qualities. If you assume that the sky is covered in material, and that all material is paisley, it's logically valid to conclude that the sky is paisley. Obviously, this isn't true.

However, sentences such as "The sky is blue" can change their truth value (it might be cloudy and grey, for example). First-Order logic cannot cope with changes; it merely describes what is actually the case at some given point in time. Nor can it cope with alternative situations ("What if...?"). The sky may be blue today, but did it have to turn out this way? These types of problems are called **contingencies**. Contingent sentences can be either true or false, depending on the situations they are evaluated in.

Examples 1: Which of the following are contingent?

- (1) Micky Mouse is older than Michael Jackson.
- (2) I am younger than the oldest person in the world.
- (3) $2 \times 3 = 6$
- (4) All black birds are black.

For a sentence to be contingent, we need to come up with at least one possible situation in which it can be true, and one in which it can be false. We can imagine a world in which Mickey Mouse was created before Michael Jackson was born, and equally, a world in which Mickey Mouse was created after Michael Jackson was born. Similarly, we can imagine a world in which I am younger than the oldest person in the world, and one in which I am not — if, for example, I *was* the oldest person in the world. However, $2 \times 3 = 6$ no matter what happens in the world — unless, that is, the symbols themselves change their meanings. Similarly, all black birds have to be black, because if they weren't, we'd have a logical contradiction. So, examples (1) and (2) are contingent, and (3) and (4) are not.

Possibility and Necessity

First order logic cannot deal with contingent sentences, and so Modal logic was created. This introduces the notion of **Possible Worlds**. Possible worlds are really nothing more than "What if...?" descriptions. To give an example of this, you could ask, "what if the sky were a different colour?" This question would result in a set of possible worlds, of which one possible world would be exactly the same as this one, but with a grey sky; another would be identical except for having a clear sky; another might have a grey sky and be snowing; and so on.

The problem with possible worlds is that there might be an infinite number of them, and it may not be at all clear how they relate to each other. Modal logic

imposes a structure on this set of worlds through **accessibility relations**. An accessibility relationship simply provides a way of linking one world to another. So, "wait five minutes", or "change one property of an object", "remove one object from this world", or even "do nothing" would all be valid accessibility relations.

Once possible worlds and accessibility relations have been defined, Modal logic sets about specifying what it means for something to be possible. A statement is **possible** (formally written as " \Diamond ") if it is true in at least one accessible possible world. From this definition, the notion of "**necessity**" can then be derived. A statement is **necessary** if it is not possible for it to be false ($\sim\Diamond\sim p$) — in other words, that it can't not happen. What this rather confusing definition actually means is that a statement is **necessary** (formally written as " \Box ") if it is true in every accessible possible world. Once these two terms have been defined, Modal logic can then deal with the sentences first order logic could not, since contingent statements are possible, and non-contingent statements are either necessarily true or necessarily false.

Examples 2: Are the following statements possibly true, necessarily true, or necessarily false?

- (1) Micky Mouse is older than Michael Jackson.
- (2) I am younger than the oldest person in the world.
- (3) $2 \times 3 = 5$
- (4) All black birds are black.

Looking at exercise 2, we can say that (1) and (2) are possible truths, since they are contingent. (3) is necessarily false, since there are no situations when it could possibly be true, and (4) is necessarily true, since (as before) there are no situations when it could possibly be false.

Modal Systems

The choice of accessibility relation is extremely important, since each works in its own way. For example, the relationship "change one property of an object" allows you to return to your initial possible world (you just change the property back), but "remove one object" doesn't. As you progress, you can see that the "remove one object" relationship will eventually reach a world in which nothing exists, no matter what route you take to get there. There is nothing comparable to this terminal state for the relationship "change one property of an object". This means that from the same starting point (i.e. the same original possible world), a statement which is possible with one accessibility relationship might be impossible (necessarily false) with another, and must be the case (necessarily true) with a third relationship.

Fortunately, many accessibility relationships behave in similar ways. This enables us to classify them into systems of Modal logic according to their properties. These properties will often be things like, "you can return to a previous state", or, "you can progress as many steps along this chain of states as you like". These tell us under what conditions conclusions will be valid. If, for example, you had two accessibility relationships called A and B, "p is necessary" might be A-valid but not B-valid. This means that relationship A will have

different properties to relationship B, and that, as a result, they will belong to different Modal systems. A simple way of summarising what properties need to hold for a conclusion to be valid is to say which system it will be valid in — so, a conclusion to a proof might be that “p is necessary” is A-valid.

The three most common systems are called T, S4 and S5. These will be defined below. To demonstrate how they work, a “Grid world” will be used. These will look like this:

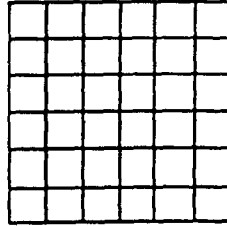


Figure 1: the grid from a “grid world”

On each grid world we can have three types of object: Circles, which are the things we will be reasoning about; Letters, which represent different objects; and walls. Circles will be treated like balls on a playing board — imagine that they can roll around to other places on the board. The only things they won’t be able to do are roll off the edges, roll on top of objects, or cross walls. An example of a grid world with all these features is shown in figure 2.

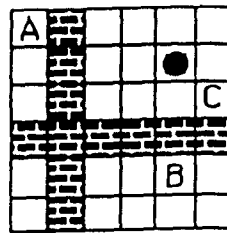


Figure 2: A grid world with one circle, two walls, and three objects

The proofs we will look at will describe relationships in this world. To avoid ambiguity, we define the relationships as follows:

- “The circle is left of X” — the circle is in a row left of the one with X in it. This means that in figure 2, the circle is left of C, but is not left of B.
- “Right of”, “Above”, “Below” — these are all defined in a similar way to “Left of”, so that they won’t be true if the circle’s just in line with a letter. This means that it’s impossible for a circle to be left of *and* right of the same object.
- “The circle is between A and B” — this will only be true if the circle is right of A and left of B (assuming A is to the left of B), *or* if it is below A and above B (assuming that A is above B). In figure 2, the circle is between A and C because it’s right of A and left of C. It’s also between A and B, because it’s below A and above B.

Now that we have something to talk about, we need something to describe the way different Modal systems work. We do this with the following conventions:

Each possible world will be represented by a grid, together with a set of statements. If a statement is false, then add, "(F)", to the end of that sentence. If the truth value of a sentence isn't known yet, add "(?)". An example of a possible world is given in figure 3.

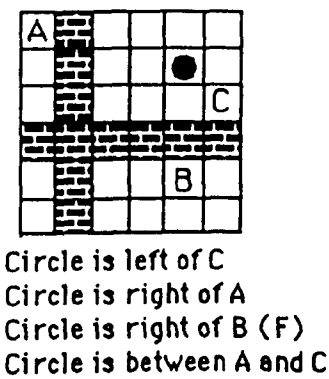


Figure 3: A possible world, with a grid and sentences.

To show that two possible worlds are related, we draw an arrow from one to the other. The direction of the arrow shows which possible world can be reached from the other. If there are other possible worlds which we don't yet need to consider, we show this with a second arrow, at the end of which is just a box saying, "Etc." This saves having to work out all the different worlds which are possible unless we actually need to do so. Figure 4 shows part of the relationship "the circle can move one square".

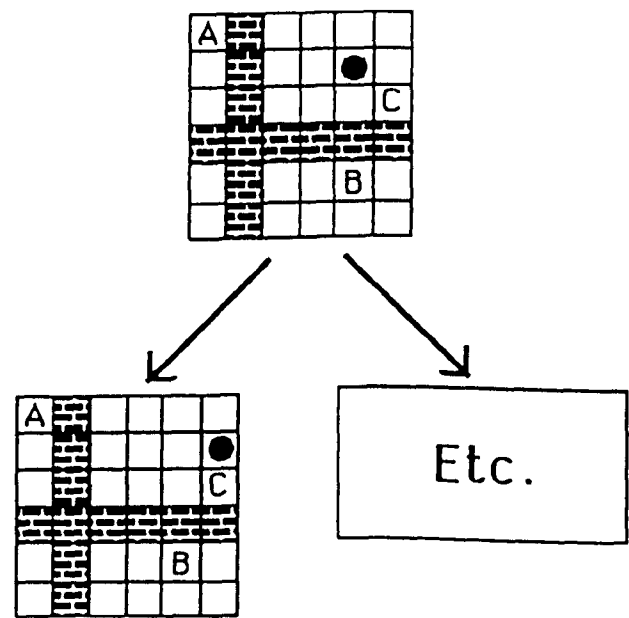


Figure 4: An example of a relationship between possible worlds

Now that we have defined all the essential elements of this type of logic (possible worlds, and the relationships between them), we can start

constructing Modal proofs. A Modal proof is a completed diagram which investigates whether some claim is valid or not. If the diagram cannot be completed, or contains contradictions, then the statement is invalid. If it can be completed without contradictions, it is proved valid.

To demonstrate how a proof can be constructed, we will work through an example. Let's say we want to investigate the claim, "the circle must be between A and B" for the world shown in figure 5. We also need to establish explicitly how the things in this world will behave. In this case, walls and objects cannot move, and that the circle can be pushed so that it will roll one square vertically or horizontally. It cannot be pushed more than once, but we can choose not to push it, so that the situation remains unchanged.

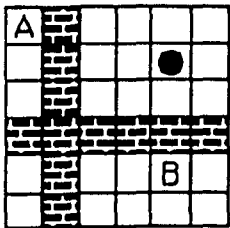


Figure 5

Can we show that this is a valid claim? It is often easier to show that a claim is invalid, because we only need construct one counter-example in such a case. To prove that a claim is valid, we would need to construct the complete diagram, as shown in figure 6. (The loop represents the relationship "do not move the circle")

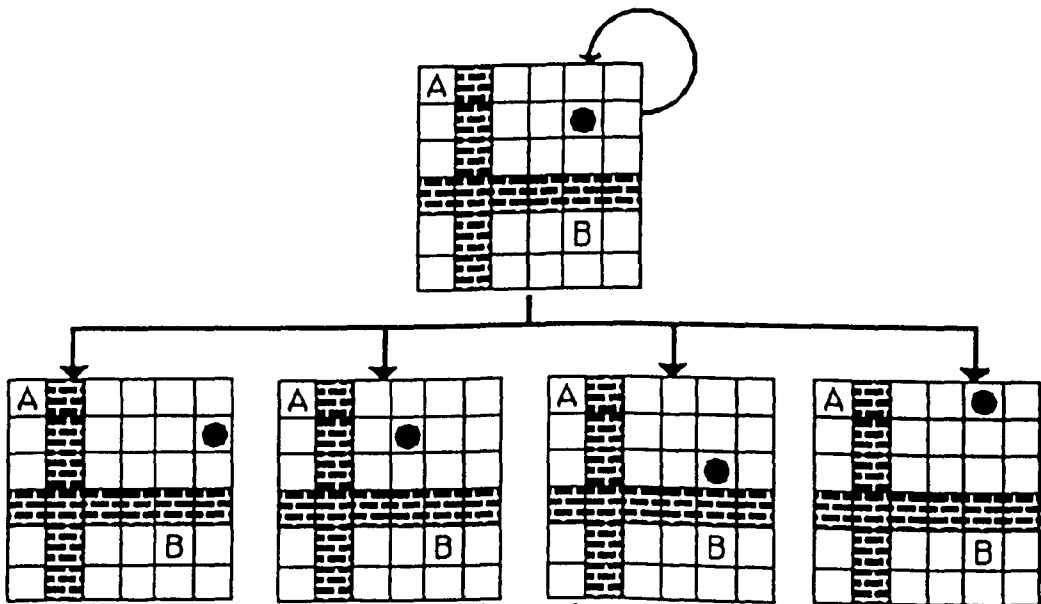


Figure 6

To show that the claim made for figure 5 is invalid, we need a diagram containing one starting world and one accessible possible world which contradicts our claim. We can leave the rest of the worlds shown in figure 6 in shorthand by using an "etc" box. Figure 7 shows just such a diagram. Here, the accessible possible world contradicts the claim made in the starting world (that the circle must be between A and B). Just by looking at the grid world, we can

see that “Below A” is not true. Because of this, “Between A and B” cannot be true either. In other words, we have constructed a diagram in which a world contradicts our initial conditions. This means that “the circle must be between A and B” is not a valid claim for the world shown in figure 5.

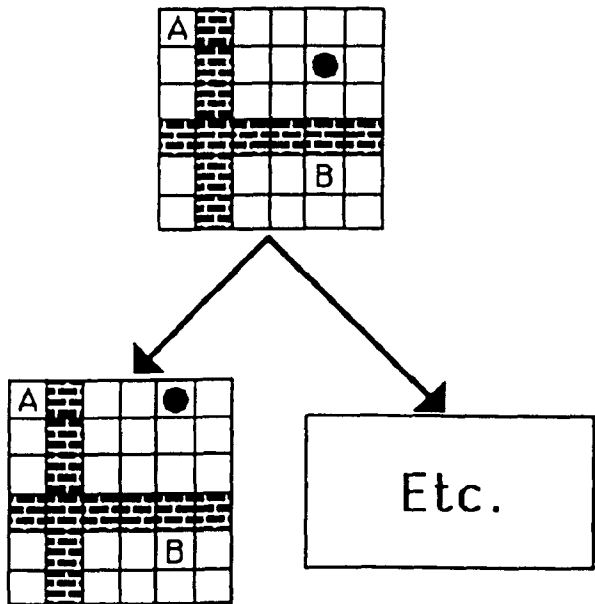


Figure 7

Trying to prove that something is the case, such as “the circle must be right of A”, is a little harder. All possible cases need to be considered, meaning that we cannot use the shorthand of figure 7 — we have to resort to completed diagrams such as figure 6. For this reason, the “etc” box will often come in handy when proving that something is possible, or when disproving that it is necessary. It will probably be of little use when proving that something is impossible, or that it is necessarily the case.

What happens, though, if we try and extend the problem? We could alter the rules, for example, so that we can move the ball twice, but add the condition that each move must take it to a new square, somewhere it hasn’t yet been. Figure 6 would then develop into figure 8, shown on the next page. Note that, to conserve space, only the top right-hand section of the accessible possible worlds are shown, since this corner covers all the places the ball could move to. The rest of the diagram remains unchanged. The relative positions of objects A and B have been marked on each.

This kind of structure — hierarchical trees, where each layer represents the “next step” — describe the first of the Modal systems, **system T**. Two things are worth noting about this system: firstly, each “layer” in the tree is treated independently — it’s as if you can see what’s coming up next, but cannot predict what might happen after that. Secondly, the branches in these diagrams don’t cross or meet. Each possible world splits to create it’s own little tree diagram, and so on to infinity, but branches aren’t allowed to be joined, even if they appear to be identical.

The effect of this is that we can “stack” Modal operators. We could, for example, try to evaluate the statement, “It must be the case after one move that the ball must be between A and B”. This is because we can investigate sentences involving Modal operators at any “level” on the tree. Say, for example, we asked of the lowest level, “must the circle be above B?” For the next level up, we could ask, “Could it be the case after one more move that the circle must be above B?” For the top level, we could do the same again, stacking another operator on top, effectively making this into a statement such as,

□◇□ “the circle must be above B”

What would have happened in our example if, instead of treating the moves separately, we asked instead, “if we allow the circle to be rolled as often as we want, and let it return to squares it has already visited, will it be the case that it must be between A and B?” In fact, these conditions describe a new system of Modal logic, the System S4. This extension to our problem introduces a new quality, transitivity — the ability to “jump” worlds, to look forward and see what will be the case in the future, not just at the next step. Effectively, system S4 is to do with asking questions such as, “will it be the case *during* this specified amount of time that such and such will be true?” Figure 9 gives an example of how this would look, starting with our usual example, and proving that “it is possible that the circle is right of B” is S4 valid.

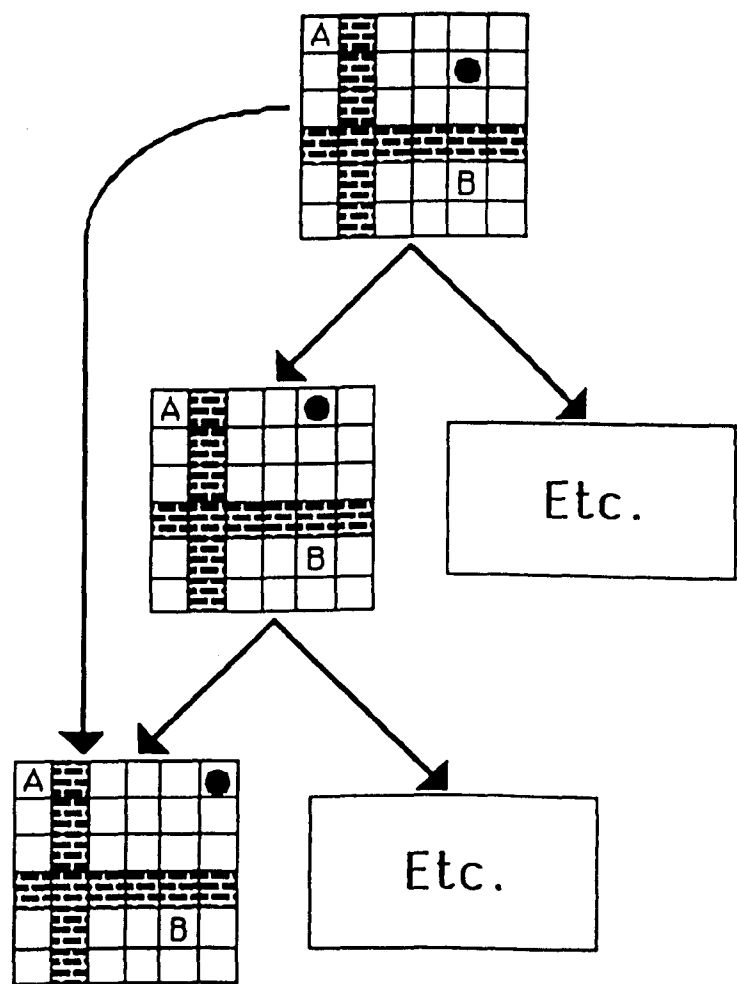


Figure 9

S4 has another important property, too. It still doesn't allow you to jump across to different branches of trees, but it does allow you to draw an arrow back upwards under certain conditions. Say, for example, your circle was moving around and around in a little pattern of four squares, for example, those shown in figure 10.

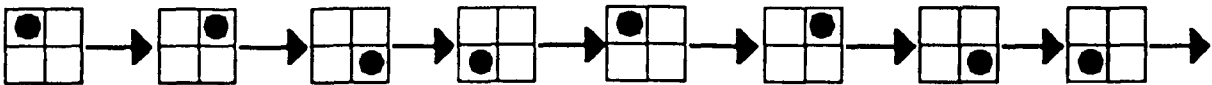


Figure 10

Instead of running this pattern on to infinity, we could draw an arrow back up to the top, so that we only needed to describe it once. It wouldn't matter how many times around the loop we went, we would still be describing the same worlds. Such a loop would look like this:



Figure 11

This makes it possible to solve problems which system T can't cope with. It also does something new to the operators. Obviously, we could have long, drawn out sentences for each of the worlds in this loop, each one stacking another operator on top of the next one, such as,

“It must be the case after the next move that it might be the case after the next move that it must be the case after the next move that it might be the case...”

and so on. Formally, this would look something like this:

$$\Box\Diamond\Box\Diamond p$$

What the loop allows us to do is cancel down *any* repeated patterns of Modal operators, something we aren't allowed to do in T. It doesn't matter how simple or complex these patterns are — usually, the easiest ones to cancel are just those which repeat the same operator twice or more. Cancellation means that in this case, $\Box\Diamond\Box\Diamond p$ would simply become $\Box\Diamond p$.

In summary, S4 tackles a slightly different set of problems to T, allowing us to analyse not just what will happen *after* a certain amount of time, but also what will happen *during* that time.

System S5 is another modification of the original system, T. This looks at even more general claims than S4, such as, “can such and such *ever* be the case?” Going back to our example, we could ask, “must the circle *always* be between A and B?”

The effect of this is that the structure of the diagrams changes again. Here, every world must be accessible to every other one. Instead of asking, “what will happen if we allow the circle to move once?”, we need to look at “where could the circle eventually get to from here?” In effect, we have done away

different branches on the tree diagram, and simply reached a stage where every world has an arrow pointing to every other world. For our example, we will need one world for each of the squares in the right hand corner of the grid, as these are the only ones that the circle can roll to. This would give a diagram that looks like figure 12. (Note that the arrow heads have been removed, so that you can travel either way along each line. Otherwise, there would need to be twice as many lines drawn)

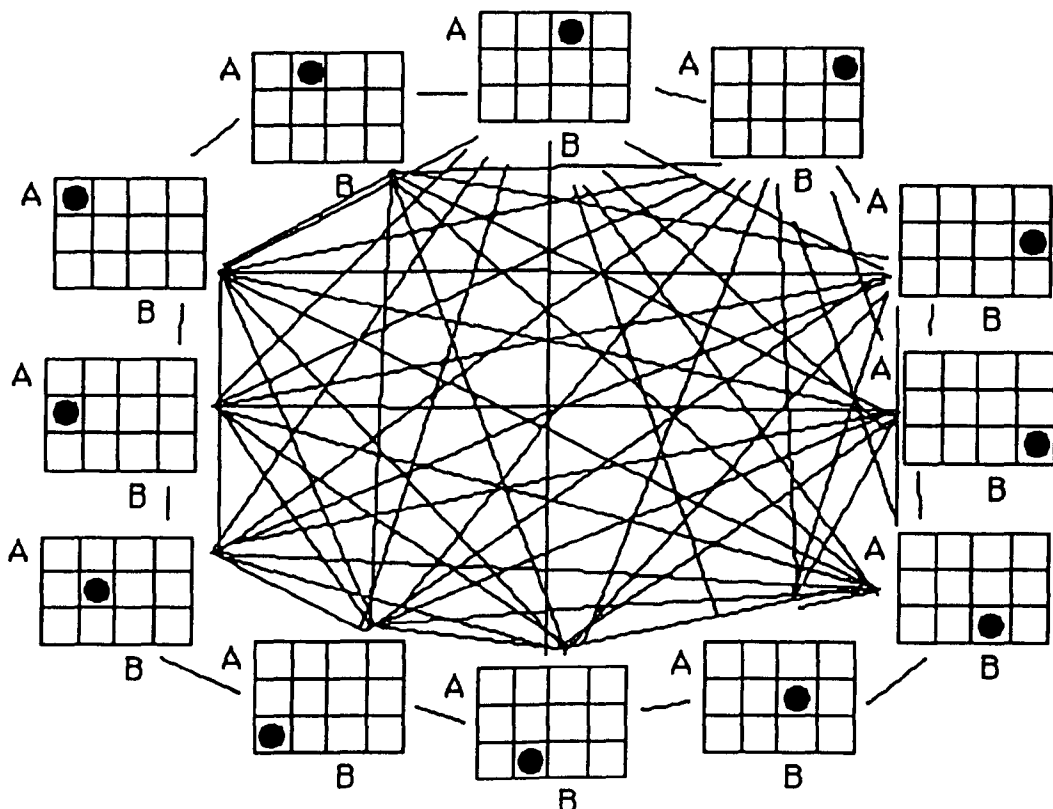


Figure 12

However, since it becomes almost impossible to make sense of this kind of diagram once more than four or five states are involved, we can adopt a simpler representation instead. Just as we could in S4, S5 allows us to jump forward along as many arrows as we like. Consequently, we can just omit all but one series of connecting arrows, making a note that we can use more than one at once (Figure 13).

We saw with S4 that altering the rules of making diagrams allowed us to cancel down Modal operators. The same is true here, but in a much more drastic form. If something is necessary in one world, it must be true in each and every world (since they're all accessible from any one given world). Similarly, if something is possible in one world (call it X), there must be at least one world it can reach where that thing is true (call it Y). That means this thing must be possible in every world, since all the other possible worlds can also reach Y, too.

In effect, this means that any "stack" of Modal operators can be treated like a loop and cancelled. In S5, then, stacked operators can always be cancelled down, leaving only the last operator. So something like $\Box\Diamond\Box\Diamond p$ would simply become $\Diamond p$

A table of examples of the different system's effects on stacked Modal operators is given below.

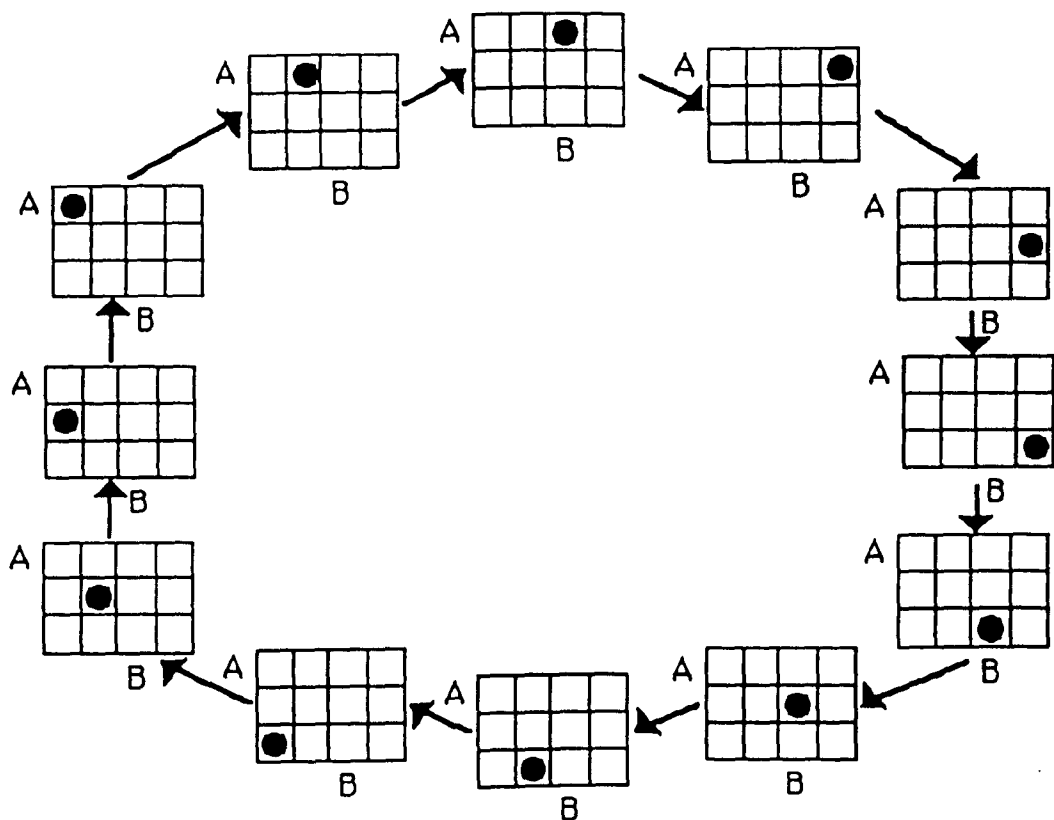


Figure 13: Travel along as many arrows as desired.

Examples 3: Operators in different modal systems

Formulae	In system T	In system S4	In system S5
$\Box\Box p$	$\Box\Box p$	$\Box p$	$\Box p$
$\Box\Diamond\Box p$	$\Box\Diamond\Box p$	$\Box\Diamond\Box p$	$\Box p$
$\Box\Diamond\Diamond p$	$\Box\Diamond\Diamond p$	$\Box\Diamond p$	$\Diamond p$
$\Diamond\Box\Box\Diamond\Diamond\Diamond\Box\Box p$	$\Diamond\Box\Box\Diamond\Diamond\Diamond\Box\Box p$	$\Diamond\Box p$	$\Box p$

Time Finished:

Post-test: Section A

Name:

Date:

Question 1

For each of the following sentences, say whether they could happen or not. Briefly explain why.

- a) London Bridge is in Louisiana.
- b) I am older than my brother, who is older than my sister, who is older than me.
- c) I believe that $3 \times 2 = 5$.
- d) $3 \times 2 = 5$

Question 2

Say whether each of these sentences is contingent or not (i.e. that their truth values can change). Briefly explain why.

- a) All viola players are musicians.
- b) All Redcurrants are green.
- c) All red currants are green.
- d) All people from England are English.

Additionally, say which of these four are necessarily false, possibly true, or necessarily true.

Question 3

Say when, if ever, each of the following are valid. You may use the names T, S4 and S5 to summarise these conditions if you wish. 'p' denotes any proposition.

You may use either diagrams or symbols in your solutions.

- a) $\Diamond\Diamond p \rightarrow \Diamond p$
- b) $\Diamond p \rightarrow \Diamond\Diamond p$
- c) $\Box\Diamond\Box\Diamond\Box p \rightarrow \Box\Diamond\Box p$
- d) $\Diamond(p \ \& \ q) \rightarrow (\Diamond p \ \& \ \Diamond q)$

Question 4

Let I be a Modal system with the following property:

In I, each world is only related to itself. In other words, to calculate the truth value of a modal statement (such as "it is possibly true that..."), only the truth values of the actual world are considered.

For system I, show that $p \leftrightarrow \Box p$.

Give an everyday example of when such a system could occur.

Post-Test: Section B

- 1) How do you think you did with the tests?
- Very well..... ☐
- Fairly well ☐
- Ok ☐
- Poorly..... ☐
- Very badly ☐
- 2) Did you enjoy it at all?
- Yes, it was fun ☐
- It was interesting, but not really enjoyable..... ☐
- Neither liked nor disliked it..... ☐
- Not really ☐
- Not at all..... ☐
- 3) Did you have any problems with question 1 (d) on the tests? (e.g. “ $3 \times 2 = 5$ ” - can this happen?) Could you explain a bit about what the problem was?
- 4) Did you have any problems with question 3 on the tests? (e.g. when is “ $\square \diamond \square \diamond \square p \rightarrow \square \diamond \square p$ ” valid?) Could you explain a bit about what the problem was?
- 5) Did you have any problems with question 4 on the tests? (When a new type of system is defined) Could you explain a bit about what the problem was?

- 6 a) Were any of the ideas covered in the material particularly difficult?
Please tick which ones from this list you are still unsure about:
- What is meant by “must” and “might” ☐
 - Problems of reference (e.g. could $2 \times 3 = 5$?)..... ☐
 - Contingency..... ☐
 - Accessibility Relations..... ☐
 - What the modal operators (\Box and \Diamond) mean, and how they
are used..... ☐
 - Constructing Modal proofs ☐
 - The differences between Modal systems (T, S4, S5)..... ☐
- 6 b) Could you explain any of these difficulties in greater detail?

- 7) What did you think about the way the ideas were represented? How could this be improved? (For example, the three conditions used in this study were just syntax, diagrams, and diagrams with simple examples — which would have suited you best and why?)

Appendix 2: Quantitative data, first study

This appendix contains quantitative data from the study described in chapter 3. These data are coded as follows:

Condition:	1 for syntactic, 2 for diagrammatic, 3 for learning environment.
Admin.:	(How the test was administered) 0 for observed, 1 for postal.
Gender:	0 is male, 1 is female
FOL Experience:	(Prior experience with first-order logic) 0 for none, 1 for some awareness, 2 for familiarity (know what the symbols mean), 3 for competent (e.g. have completed a first course in logic), 4 for extensive (e.g. have studied an advanced logic).
NSL Experience:	(Prior experience with non-standard logics) 0 for none, 1 if subject has studied Modal logic, 2 if subject has studied another non-standard logic, 3 if both Modal logic and another non-standard logic has been studied.
Study time:	Length of time taken to work through the instructional material, to the nearest 5 minutes.
Pre World:	(Pre-test understanding of possible worlds) 0 for not mentioned, 1 for used correctly, 2 for an awareness of variants on the real world ("if this happened..." answers), 3 for responses suggesting that only the real world is possible.
Post World:	(Post-test understanding of possible worlds) Coding as Pre World.
Pre Belief:	(Pre-test understanding of belief) 0 for no answer, 1 for used correctly, 2 not understood.
Post Belief:	(Post-test understanding of belief) Coding as Pre Belief.
Pre Ref:	(Pre-test understanding of problem of reference) 0 for no awareness, 1 for aware only that names can change reference, 2 for aware only that symbols can change reference, 3 for awareness of both names and symbols changing reference.
Post Ref:	(Post-test understanding of problem of reference) Coding as Pre Ref.
Pre Poss:	(Pre-test classification of statements according to their possibility or necessity) Scored from 0-4 on the number of correctly justified classifications.
Post Poss:	(Post-test classification of statements according to their possibility or necessity) Coding as Pre Poss.
Pre Syst:	(Pre-test understanding of Modal systems) 0 for no answer, 1 for only answering 'valid' or 'invalid', 2 for problems with system T, 3 for problems with both S systems, 4 for problems with S4, 5 for problems with S5, 6 for complete understanding of Modal systems, and 7 for proofs attempted without any reference to Modal systems.
Post Syst:	(Post-test understanding of Modal systems) Coding as Pre Syst.
Pre proof:	(Pre-test ability to construct Modal proofs) 0 for no correct proofs, 1 for some correct proofs, 2 for all answers having a correct proof.
Post proof:	(Post-test ability to construct Modal proofs) Coding as Pre proof.
pre-Q3 score:	Pre-test score on question 3, ranging from 0 to 12.
Post-Q3 score:	Post-test score on question 3, ranging from 0 to 12.
Q3 Improved	A comparison between subjects' pre- and post-test scores on question three. The symbol '/' indicates that the subject failed to attempt the question at either pre- or post-test.
How I Scored:	0 for very badly, 1 for badly, 2 for ok, 3 for well, 4 for very well.
Enjoyment:	Subjects' self-assessed enjoyment, ranked from 0 (hated it) to 4 (enjoyed it).
Pref. for type:	1 if subject preferred syntactic material, 2 if subject preferred diagrammatic material, 3 if subject preferred learning environment material, and 4 if the subject preferred a combination of the learning environment and syntactic materials.
Pre-Q4 description	(Whether the subject showed demonstrated an understanding of the system described in the question) 0 indicates that the subject didn't attempt the

	question, 1 denotes that they showed an understanding, 2 means that they attempted the question but failed to understand the system.
Post-Q4 description	Coding as pre-Q4 description.
Pre-Q4 proof	0 means that the subject made no attempt to answer the question, 1 shows that they attempted the question, but had a problem understanding the symbols used, 2 means that the subject constructed a successful diagrammatic proof but didn't attempt a syntactic one, 3 means that the subject constructed an unsuccessful diagrammatic proof and didn't attempt a syntactic one, 4 means that the subject constructed a successful syntactic proof but didn't attempt a diagrammatic one, 5 means that the subject constructed an unsuccessful syntactic proof and didn't attempt a diagrammatic one, and 6 indicates that the subject gave successful syntactic and diagrammatic proofs for the claim.
Post-Q4 proof	Coding as pre-Q4 proof.
Pre-Q4 Example	(Whether the subject was able to give an example illustrating the system described in the question) 0 indicates that the subject didn't attempt the question, 1 denotes that they gave a correct example, 2 means that they attempted the question but failed to give a correct example.
Post-Q4 Example	Coding as pre-Q4 example.
Self: Operators	(Subjects' self-assessed problems with the concept of Modal operators) 0 for no answer, 1 if understood this concept, 2 if concept was problematic.
Self: Reference	(Subjects' self-assessed problems with the problem of Reference) 0 for no answer, 1 if understood this concept, 2 if concept was problematic.
Self: Systems	(Subjects' self-assessed problems with the concept of Modal systems) 0 for no answer, 1 if understood this concept, 2 if concept was problematic.
Self: Proofs	(Subjects' self-assessed problems with the concept of Modal proofs) 0 for no answer, 1 if understood this concept, 2 if concept was problematic.
Self: Acc. Rel'n	(Subjects' self-assessed problems with the concept of Accessibility Relations) 0 for no answer, 1 if understood this concept, 2 if concept was problematic.

Subject	Condition	Gender	Admin	FOL	NSL	Study time
1	1	0	0	1	0	25
2	1	1	1	0	0	25
3	1	0	1	3	0	30
4	1	1	0	1	0	10
5	1	0	1	2	0	20
6	1	1	0	0	0	20
7	1	1	0	0	0	25
8	1	0	0	1	1	30
9	1	1	1	1	0	40
10	1	0	1	0	0	45
11	1	0	1	3	3	20
12	1	1	0	2	0	25
13	1	1	1	3	1	10
14	1	0	0	2	0	20
15	1	0	1	1	0	15
16	1	0	1	2	0	45
17	2	0	1	3	1	50
18	2	1	1	0	0	20
19	2	0	0	3	0	35
20	2	1	1	3	0	40
21	2	1	1	3	2	60
22	2	1	1	0	0	25
23	2	1	0	1	3	35
24	2	0	0	0	0	45
25	2	0	0	0	0	10
26	2	0	1	2	0	20
27	2	1	0	2	0	20
28	2	1	0	0	0	20
29	2	0	1	1	0	35
30	2	1	1	1	0	35
31	2	0	0	1	0	45
32	3	1	0	1	0	45
33	3	0	0	1	0	35
34	3	0	1	1	0	10
35	3	1	1	0	0	10
36	3	0	1	3	0	30
37	3	1	1	3	0	30
38	3	1	0	0	0	30
39	3	1	0	1	0	35
40	3	0	1	3	0	40
41	3	1	1	1	0	50
42	3	0	1	4	1	20
43	3	1	0	1	0	50
44	3	0	0	2	3	20
45	3	1	1	0	0	20
46	3	0	1	3	0	15
47	3	1	0	2	3	40

Subject	Condition	Pre-Worlds	Post-Worlds	Pre-Beliefs	Post-Beliefs	Pre-Reference	Post-Reference
1	1	3	2	2	1	0	0
2	1	2	2	1	1	0	0
3	1	3	3	2	1	0	0
4	1	2	1	1	1	3	0
5	1	1	1	1	1	1	1
6	1	2	3	1	1	3	2
7	1	3	1	2	2	0	0
8	1	3	1	1	1	2	3
9	1	3	1	1	1	0	3
10	1	3	2	2	1	3	2
11	1	3	1	2	1	0	1
12	1	2	1	2	1	2	2
13	1	1	0	1	1	3	3
14	1	3	2	2	1	0	0
15	1	1	1	1	1	3	3
16	1	1	1	1	1	1	3
17	2	3	1	2	1	0	3
18	2	2	1	1	1	3	3
19	2	3	2	1	1	2	0
20	2	1	1	1	1	2	3
21	2	3	1	1	1	0	1
22	2	2	1	1	1	0	1
23	2	3	2	2	2	0	3
24	2	3	1	2	1	0	0
25	2	3	3	2	2	0	0
26	2	1	1	1	1	1	1
27	2	3	3	1	1	0	0
28	2	3	2	2	2	0	3
29	2	2	1	1	1	2	1
30	2	1	3	0	1	3	3
31	2	3	2	0	1	0	0
32	3	3	2	2	1	0	0
33	3	3	3	2	2	2	2
34	3	3	1	1	1	0	1
35	3	2	1	1	1	2	1
36	3	0	3	1	1	0	2
37	3	2	0	1	1	2	2
38	3	1	1	1	1	3	1
39	3	3	0	2	1	2	2
40	3	3	1	2	2	0	2
41	3	1	1	2	1	3	3
42	3	1	1	1	1	1	3
43	3	2	2	2	1	0	1
44	3	1	1	1	1	1	1
45	3	2	1	1	1	0	1
46	3	3	3	2	2	0	2
47	3	1	1	1	1	3	1

Subject	Condition	Pre- Poss	Post- Poss	Pre- Systems	Post- Systems	Pre- Proofs	Post- Proofs
1	1	3	3	7	6	1	1
2	1	2	4	7	6	0	1
3	1	4	4	1	6	0	1
4	1	3	4	0	0	0	0
5	1	4	4	0	6	0	0
6	1	4	2	0	0	0	0
7	1	2	1	0	0	0	0
8	1	1	4	1	6	0	0
9	1	2	4	7	0	0	0
10	1	2	4	0	6	0	0
11	1	3	4	1	5	0	0
12	1	4	4	0	5	0	0
13	1	4	4	1	6	0	0
14	1	2	4	0	6	0	1
15	1	4	4	0	6	0	0
16	1	4	4	0	6	0	2
17	2	4	4	7	4	2	1
18	2	4	4	1	1	0	0
19	2	4	4	1	4	0	1
20	2	2	3	1	0	0	0
21	2	1	1	2	3	1	0
22	2	3	4	7	0	0	0
23	2	1	4	7	6	1	1
24	2	3	4	1	6	0	1
25	2	3	4	0	0	0	0
26	2	3	4	0	7	0	1
27	2	1	3	0	0	0	0
28	2	0	1	0	0	0	0
29	2	3	4	0	0	0	0
30	2	3	4	1	1	0	1
31	2	0	2	0	0	0	0
32	3	4	4	0	2	0	0
33	3	2	4	7	6	0	0
34	3	4	4	1	1	1	0
35	3	3	1	0	0	0	0
36	3	4	4	0	0	0	0
37	3	4	3	0	0	0	0
38	3	2	2	0	0	0	0
39	3	4	3	0	0	0	0
40	3	2	4	1	7	1	1
41	3	3	4	0	0	0	0
42	3	4	4	6	6	2	2
43	3	4	4	7	6	0	1
44	3	3	4	1	6	0	2
45	3	4	4	7	6	0	2
46	3	3	3	7	6	0	1
47	3	2	4	1	6	0	2

Subject	Condition	pre-Q3	post-Q3	Q3 Improved	How I scored	Enjoyment	Pref. for type
1	1	10	6	-4	1	1	4
2	1	5	3	-2	1	4	3
3	1	7	6	-1	1	1	3
4	1	0	0	/	0	0	3
5	1	0	6	6	2	3	3
6	1	0	0	/	0	0	0
7	1	0	0	/	1	0	0
8	1	6	7	1	1	4	0
9	1	0	0	/	2	4	0
10	1	0	5	5	1	3	0
11	1	6	8	2	2	3	3
12	1	0	9	9	1	1	3
13	1	10	12	2	3	3	3
14	1	0	9	9	3	3	3
15	1	0	10	10	2	3	3
16	1	0	10	10	1	3	1
17	2	9	5	-4	1	3	3
18	2	10	7	-3	2	3	4
19	2	8	5	-3	1	3	3
20	2	6	0	-6	1	4	0
21	2	4	0	-4	1	2	0
22	2	6	0	-6	1	3	1
23	2	6	5	-1	0	3	3
24	2	8	3	-5	0	0	3
25	2	0	0	/	1	3	3
26	2	0	0	/	1	3	1
27	2	0	0	/	1	1	3
28	2	0	0	/	1	3	3
29	2	0	0	/	1	3	3
30	2	9	8	-1	1	3	3
31	2	0	0	/	0	3	3
32	3	0	3	3	2	4	3
33	3	4	4	0	2	4	3
34	3	9	10	1	3	3	4
35	3	0	0	/	1	0	0
36	3	0	0	/	0	3	0
37	3	0	0	/	1	4	3
38	3	0	0	/	0	1	3
39	3	0	0	/	1	3	0
40	3	9	8	-1	3	2	4
41	3	0	0	0	0	1	4
42	3	12	12	0	3	4	3
43	3	9	10	1	2	4	3
44	3	6	8	2	2	4	3
45	3	3	9	6	1	3	3
46	3	0	7	7	2	3	3
47	3	4	12	8	3	4	3

Subject	Condition	pre-Q4 description	post-Q4 description	pre-Q4 proof	post-Q4 proof	pre-Q4 example	post-Q4 example
1	1	0	1	0	5	0	2
2	1	2	0	5	0	2	0
3	1	0	1	0	4	0	0
4	1	0	0	0	0	0	0
5	1	0	0	0	0	0	0
6	1	0	0	0	0	0	0
7	1	0	0	0	0	0	0
8	1	0	1	0	4	0	0
9	1	0	0	0	0	0	0
10	1	0	0	0	0	0	0
11	1	0	0	0	0	0	0
12	1	0	0	0	0	0	0
13	1	0	1	0	4	0	1
14	1	0	1	0	4	0	1
15	1	0	0	0	5	0	0
16	1	0	1	0	4	0	0
17	2	2	2	5	5	0	0
18	2	0	1	0	0	0	1
19	2	0	0	0	0	0	0
20	2	0	0	0	0	0	0
21	2	0	0	0	0	0	0
22	2	0	0	0	0	0	0
23	2	0	1	0	5	0	2
24	2	0	0	0	0	0	0
25	2	0	0	0	0	0	0
26	2	0	2	0	3	0	2
27	2	0	0	0	0	0	0
28	2	0	0	0	0	0	0
29	2	0	0	0	0	0	0
30	2	0	1	0	4	0	0
31	2	0	2	0	0	0	0
32	3	0	0	0	0	0	0
33	3	0	0	0	0	0	0
34	3	2	2	5	0	0	2
35	3	0	0	0	0	0	0
36	3	0	0	0	0	0	0
37	3	0	0	0	0	0	0
38	3	0	0	0	0	0	0
39	3	0	0	0	0	0	0
40	3	0	1	0	2	0	1
41	3	0	0	0	0	0	0
42	3	1	1	6	6	1	1
43	3	0	1	0	2	0	1
44	3	2	0	5	0	0	0
45	3	0	0	0	0	0	0
46	3	0	1	0	0	0	1
47	3	0	1	0	1	0	1

Subject	Condition	Self: Operators	Self: Reference	Self: Systems	Self: Proofs	Self: Acc. Rel'n
1	1	2	2	2	2	0
2	1	1	1	2	2	0
3	1	1	1	2	2	2
4	1	1	1	2	2	0
5	1	1	1	1	2	1
6	1	2	1	2	2	2
7	1	2	1	2	2	2
8	1	1	2	1	2	1
9	1	2	2	2	2	2
10	1	2	1	2	2	1
11	1	1	1	2	1	2
12	1	2	1	1	2	1
13	1	1	1	1	2	1
14	1	1	1	1	1	0
15	1	1	1	2	2	1
16	1	1	1	2	2	2
17	2	2	1	2	2	2
18	2	2	1	2	2	1
19	2	1	2	1	2	0
20	2	1	1	2	2	2
21	2	2	1	2	2	1
22	2	2	1	2	2	2
23	2	1	2	2	2	0
24	2	1	2	1	2	0
25	2	2	1	2	2	1
26	2	1	1	1	2	1
27	2	2	2	2	2	2
28	2	2	1	2	2	2
29	2	2	1	2	2	2
30	2	2	1	2	2	1
31	2	2	2	1	2	0
32	3	1	2	2	2	0
33	3	2	1	2	1	1
34	3	1	1	2	1	1
35	3	2	1	2	2	2
36	3	2	1	2	2	2
37	3	1	2	2	2	1
38	3	2	1	2	2	2
39	3	2	1	2	2	1
40	3	1	1	2	2	2
41	3	2	0	2	2	2
42	3	1	1	1	1	1
43	3	1	1	1	2	0
44	3	1	1	2	2	0
45	3	2	1	2	2	1
46	3	1	1	1	1	1
47	3	1	1	1	1	1

Appendix 3: Qualitative data, first study

This appendix contains comments from the open responses to post-test questions for the study described in chapter 3.

The abbreviated codes indicates that the following comment relates to the following:

Studied:	where logic was studied
q3:	question 3
MOp:	the use and understanding of Modal operators
Transfer:	the near transfer question
Ref:	the problem of reference
Sys:	the use and understanding of Modal systems
Proof:	constructing Modal proofs
Style:	the representational style of the subject's condition
Pref:	a preference for one of the three representational styles
Other:	miscellaneous comments not contained under other headings

Syntactic condition

Subject 1

MOp:	I understood them when there was context, a sentence that they belonged to, for example something is necessary for something else, something might be necessary for something else, but strings of symbols were horrible.
Ref:	Erm... it was mentioned in there, but it seemed as though for everything you could just change the meaning, you could make black white. So I assumed that we were working in base ten...
Sys:	I understood some of the simple differences, and how they were building up, transitive, and that... I didn't quite fully understand how they were meant to fit in the proofs.
Style:	Complicated. [Interviewer prompts] I don't think the average person's going to pick up on the algebraic expression of the argument. I just don't think... I think they might pick up on it with some relevant contextual... kind of...
Pref:	[Diagrammatic] I mean, comparing that with the algebraic one. Well, I haven't looked at that one in that much detail, but that one seems, you know, a bit easier to follow. [Interviewer prompts] They'd both need a kind of text to go with it. But um... I don't think anyone would ever use that.
Other:	It was a frightening reminder of what degree level text books were like towards the end.

Subject 2

Transfer:	I think this bit confused me so much that when I looked at it, I thought, no I can't do this.
Sys:	Sort of. [Interviewer prompts] It's just too mathematical. I think one needs to spend a lot of time working out more and more examples of this before you do the test.
Pref:	[Diagrammatic] I think this one is a bit easier, it's just a very superficial impression, for someone with a non-mathematical background, because the other one's a bit intimidating. So this seems a bit more accessible.
Pref:	[Diagrammatic and Syntactic] I think the two together would work well, because that would reach more kinds of people.

Pref: [Diagrammatic] To mean this makes it a bit clearer, because it's more visual. The other one is not obvious.

Pref: [Learning Environment] Oh, that is clever, actually. Yes, this is the most real of the lot. It's the easiest to imagine if you've got a non-scientific mind.

Other: I think I've ended up getting more confused by it. The symbols bit of it at the nodes. But the rest of it was ok.

Other: And I had more of an idea of what some of the symbols were supposed to mean, but I had less of an idea of what they actually meant. [Interviewer prompts] I am sure the biggest factor was that fact that I didn't have very much time to run through the notes, and I'm sure that if I had had more time it would have been amply clear.

Subject 3

Studied: Philosophy, Introduction to Logic

Transfer: Completely lost me. Seemed obvious?!

Style: More examples

Subject 4

Ref: I quite liked doing the word ones, but that's a little bit like, erm, party games, so I can see the fun of that.

Sys: And then the other, I mean, you further got into the symbols, I thought I understood the first bit where it said you could stack, stack them up. Um... but then I thought when I got to the second system that I didn't, because things were kind of, symbols were appearing and disappearing in a way that didn't make any sense to me at all. I couldn't follow why they were appearing or disappearing from sight on one side of the equation.

Style: No. Well, I know that I don't have any background in doing those kind of equations with symbols and things, and, well, I don't think I'm maths-phobic, but they worry me, they make me anxious, I don't like to feel there's something I really can't get my head around.

Style: Well, first of all, I didn't understand the argument about you could do anything except take things away, so you could have any system except a system in which you took things out, in which case you'd end up with nothing, because I thought the reverse of taking things out is putting them back, so why can you not have a system that does that?

Pref: [Diagrammatic] I think... I mean, you've now explained to me that there's some pattern between these systems, if I could hold on to that, then I might be better able to see what's happening in the systems, um... I don't find these boxes any more friendly than the other way. I feel that I... I need to have them talked through, symbol by symbol, do you know what I mean? Because I look at them and it is like, um, hieroglyphs or something, and I want to call them something, I don't want to call them "square p diamond q" because then that's just confusing, I want to call them something else. [Learning environment] Great. I don't have any problem with three dimensional, two dimensional representation, I have a problem with abstraction like that. I mean, I may be thinking, there's this thing, you know Matt Gross upstairs always used to say that people like diagrams because they think they understand them, they actually don't, I mean, I may be kidding myself, and maybe I don't understand them, but I think I do, which immediately makes me feel much more comfortable than looking at that and thinking that I don't!

Other: Why is it called S4?

Other: It's one of those things that I think I'm understanding when you're speaking, but when you stop speaking... [laughs] I don't think I've understood at all!

Other: Um... I don't know if it would help to give some verbal examples of what these... but you wouldn't need to if you were doing that.

Other: God... difficult stuff! I'm glad I don't *need* to know it...

Subject 5

Studied: Computer Science

q3: Yes, in the pre-test, as I did not have an understanding of how symbols relate.

Transfer: I didn't have sufficient understanding of the Modal systems for proofs.

Style: A more graphical representation would aid better understanding.

Subject 6

- Studied: That's where I've come across it. [Interviewer prompts] In artificial intelligence. Formal semantics. But that's doesn't help because I didn't understand it then, either.
- Other: I think I have a problem actually because to me Modal means something completely different. [Interviewer prompts] Well, I'm from a linguistics background, applied linguistics, so it's a different sort of Modal as in can/may/might/must. It's one of those things that similar enough to be irritating, but different enough not to make sense.
- Other: Are you talking about a prediction there, or an actual event?
- Other: My problem is with the time reference.
- Other: I can cope with this sort of thing much easier if I know why, if I have a reason for doing it. It doesn't mean I don't want to do it, but at least I can cope with it.

Subject 7

- Style: Just basically the symbols, I mean it didn't mean anything to me, and reading the instructional material it still didn't make it any clearer because I'm not that like-minded. Symbols have to mean something and I didn't grasp what was it was that was meant by those symbols so I could give them a value and couldn't put a meaning to them.
- Style: I found it... not long winded, but I found it deep and difficult to follow. It would need to be unpacked a great deal, and put in more simplistic terms, to make sense of that.
- Style: Even visually, the way it's displayed, mindset: I can't cope with that; rejection.
- Pref: [Diagrammatic] No. Because again, automatically, mindset: that, I'm not going to take in. I know I'm not. So I'm not even going to attempt it.
- Pref: [Learning Environment] That makes more sense to me. It does, yes. [Interviewer prompts] Because there's clear instructions; visually, you can unpack it quicker. You can see from the various diagrams what the objective is, what it is you're being asked to do, whereas one I could not decipher what that was about.
- Pref: I would prefer this model [Learning Environment], because visually, it appeals. I can unpack it. I can... I would make a reasonable go at that, other than that. Um... Why? Because visually, you actually can clearly see what is being instructed, what the aim objectives are, what it is you have to arrive at, or what this picture is telling you about. It's a lot clearer, it's more precise. You can tell instantly. I don't know; it's just not possible for me to look at that [Diagrammatic] and put any value, and read anything into it other than a mess of symbols.
- Pref: [Learning Environment] I think you'll find that it calls for no prior knowledge, to delve into logical explanations or lengthy explanations of what one has to do. This is clearly telling you, you can relate to arrow symbols and where that dot is. You can see there's been movement. Why this movement has taken place you don't know unless you're unpacking those words in between, but overall, visually, taking it at face value, you can see what's asked for.
- Other: For me, things have to be really clear, and they've also got to mean something. Like the value — it's got to be denoted what that is. A long string of symbols just like that does not make sense. A lot of time has got to be spent unpacking it, explaining it, before I can... I'm like that with maths symbols etc.

Subject 8

- Studied: Have never studied it formally, but tried to look at textbooks on the subject and have done 'z' etc. and some prolog and temporal logic.
- q3: Well, it's a bit complicated, but I recalled that in S4 + S5 things cancelled so I guessed it was true in them.
- Transfer: They were a bit hard for me - I'm not used to proving things and I didn't spent enough time to learn the stuff properly.
- Ref: It does depend on whether you accept standard mathematical notation. Maybe this is philosophy, but I think that there is an important difference here which in turn may give misleading results.

Subject 9

- Studied: Philosophy and Professional Judgement courses
- q3: I am a words rather than a symbols person and operate best when I can look at problems in those terms.
- Transfer: Got totally lost in the explanation; might have understood if there was more time allowed to study the material, but probably needs more verbal explanation.
- Ref: Depends on whether you regard numbers as fixed absolutely or wonder if this is just a human constraint which makes sense in our world / understanding, but might possibly not be elsewhere in the universe.
- Style: These are complex ideas and I would not presume to say how they could be improved. I am not a mathematician so am probably not a good person to ask. (I have a mathematical son & I know that he certainly thinks in representational symbols, so how one responds may well depend on the sort of brain one has)

Subject 10

- q3: I really didn't understand very much of the reasoning, although I could remember the 'conclusion' as shown in example 3.
- MOp: I found it difficult to read the symbols. I had to keep looking back at the definitions. I couldn't find a way of 'saying' what NPp meant.
- Style: I would have liked more examples to illustrate meanings. I got as far as Modal systems para 2 but then would have found an example useful. And there on every new idea, expressed algebraically, I would have like to have seen illustrated by an example. From system T onwards you seemed to be assuming a familiarity with the ideas and I couldn't follow very much although I did understand the concepts reflexive, transitive, symmetric.

Subject 11

- Studied: Formal methods
- Transfer: I read the sheets once between the times stated it was not enough and as I was not supposed to go back I could not answer it. One read was not enough for me.
- Pref: Diagrams with simple examples would have helped a lot for access relations. Just textually, the ideas are too abstract. I needed some diagrams to see ideas of how the relations would work .

Subject 12

- MOp: Wait a minute, what was the diamond again? It could be the case? I find that very confusing. It could be the case, and then it could be it could be the case.
- Transfer: I couldn't be bothered. And I thought, it's only saying what's there already.
- Ref: Some of the things about being possibly true depends on how abstract you're going to get. [Interviewer prompts] That's what I mean about "depends how abstract you're going to get". If you'd given me two things and two things, then I would say they could make five things. Unless they were mice. [laughs]
- Sys: Because... it should be right, and I don't know why I crossed it out, but I suddenly decided that it would only be... That you'd only have it with one symbol if it were in S5.
- Sys: Well, if it could be the case, then in my mind, then it should always be it could be the case. I don't see... it's this stacking I don't... you need to give me a concrete example of what that is.
- Style: That's fine. But that's because I'm quite happy with it. Because that's how I'm used to seeing it.
- Pref: [Diagrammatic] That's probably clearer. [Interviewer prompts] Because I tried to explain reflexive, transitive, and symmetric, but if I'd have had that, it'd have been easier. That's what I tried to explain, but it was very hard for me to tell you in words, whereas it's easier for me to show you in a diagram. [Learning Environment] It's fairly similar to that one, really, except it's less confusing because you haven't got all this blurb here, notation. Yeah, that's even clearer than that one. [Interviewer prompts] Because you can relate it to something concrete.
- Pref: You gave me the hardest one, then!

Subject 13

- Studied: BA Philosophy Logic (predicate + propositional calculus) studied in yrs 1+2; philosophical logic yr3.
- q3: Only in retrospect when T, S4, S5 explained. I did it linguistically.
- Transfer: On pre-test I didn't understand it. On post-test I had a stab at making sense of it. I have always found the internal coherence but seeming pointlessness of this stuff rather irritating.
- Ref: Yes because I was thinking about alternative number systems. I wasn't sure at what level the question re necessary / contingent needed to be considered.
- Style: Some 'everyday examples' might have helped, as you ask for them in the tests.

Subject 14

- MOp: I didn't really understand what I was doing, really; it was just manipulating symbols in the end, and you get a bit lost in the recursive nature of it, so I'm not sure how I did.
- Ref: Yeah, it had occurred to me in the sense that, that sort of thing is related to geometry, and in geometry you could maybe change the rules.
- Ref: Yeah, well, it's a bit dodgy... it's like you know that question, the question about Snowdon and Everest, it's hard to where... I mean, basically, anything's possible, it depends how far you go, you know like at some point in the future, Everest might have worn down quite a bit.
- Proof: I didn't really have a clue about that.
- Style: I think using two shapes which are just rotated through 45 degrees, I think could lead to problems, when I was doing this bit it kind of depended sometimes on how well you'd drawn a diamond or a square.... I think something like a square and a circle, something that's got a different shape.
- Pref: [Learning Environment] Yeah, I think that would be... that would work better. [Interviewer prompts] It would be better if you could actually sort of move the ball yourself.
- Other: I think it looks like it's a necessary condition that you kind of know first order logic really to be able to deal with this, so, it's kind of a good way to check that you know it.
- Other: You know, cause I was using, I was trying to answer some of these questions using er, you know, whether I've got red socks for this one, and whether I've got red sock and weather it's cloudy, and I wasn't really sure if that was allowed or not, sort of thing.

Subject 15

- q3: I was unsure about the associativity of the operators (i.e. can "must can must must can must" be treated as "(must can) (must must) (can must)" and "(must can must) (must can must)" etc.)
- Transfer: I felt I did not have sufficient experience in this sort of logic question (i.e. the "show that" type). The fact that a new system was defined was exciting and I was disappointed that I did not have this level of experience.
- Ref: Contingency relies upon the rigidity of our rules of multiplication.
- Sys: I was ok as long as I just thought in symbols. As soon as I tried to think in sentences I naturally came unstuck.
- Pref: In the post-test, having sealed the instructional material away, my Q3 answers were based mainly on the three facts: T - cannot cancel anything! S4 - repeated patterns can be cancelled. S5 Must can p iff can p.
- Style: It could be improved by including more worked examples.

Subject 16

- Studied: Introduction to logic, Prolog
- q3: It was a bit confusing at first, the symbols and rules of S4 and S5 confused me a bit.
- MOp: My logic was never very good and some of the restrictions and relations were hard to think of how to use.
- Transfer: I was not sure of the implications of the system or how to form the proof properly.
- Pref: I thought the examples overleaf were more confusing, even though I had read through the instructional material.

Style: I would have liked examples like in this post-test Q3 and Q4 in the material more, to give more worked examples as reference.

Diagrammatic condition

Subject 17

Studied: BSc Comp Sci Degree course (formal notations)
q3: Got stuck trying to decipher diagrams and also S4 reduction via repeated symbols. Don't understand how to reduce or derive the T/F values across worlds. Don't really understand Modal logic.
MOp: Worlds and contingencies are fine but compound propositions don't read well. I can't decipher what it's trying to say.
Transfer: No idea about this one. Totally lost on how to approach an answer.
Style: Okay at first but you lost me trying to discuss accessibility—how it is done. I prefer more step by step examples. Box diagrams difficult to understand origins.

Subject 18

Ref: I thought you might have a preconceived answer that might not allow for mine.
Pref: I work better with concepts expressed in words. The use of symbols I found too similar to algebra.
Style: I was aware of my own time constraints. In these circumstances I get very annoyed with lots of symbols that otherwise I would have been quite happy to work through.

Subject 19

Transfer: I didn't think I could give an answer. [Interviewer prompts] Well a certain amount of it must be the fact that you've got the word Modal in there, and that threw me, certainly first time round. I thought, ah, this is the difficult one, that and the fact it came last. So, yes... erm... ok, I felt I had somewhat of a better grip on the concepts where you talk about it not being reflexive, again, having read the material, I felt I had perhaps a better feeling for your meaning there, but I hadn't got a point I could start from. I must say, I have never been much good at doing proper proofs.
Ref: Because it became clear to me which area you were actually looking at, and therefore, like, the slipperiness of the question changed. For example, in your examples, you were talking about Micky Mouse and Michael Jackson, and you said you could conceive a world where, the Mouse was born before Michael Jackson, or whatever. Now, that is more difficult for me to conceive than two and two making five, because it's my stock in trade to make two and two equal five. The manipulation of symbols like that is much easier than symbols that have some real-world relationship. I couldn't quite, I couldn't quite understand what you were driving at to begin with, but having read it, I thought ah yes, right, I have a better grip. [Interviewer asks if increased abstraction makes the problem of reference clearer] For me, yes. But I do notice that I'd have a very great difficulty explaining that to other people.
Sys: I was clear there were differences, and certainly... I didn't distinguish that... I wouldn't care to tell you what the differences were.
Style: I had to go back over it several times, it was an awful lot denser than the average OU text, let's put it that way. The amount you're trying to absorb from it...
Style: If you go the other way, and say right, here's this one, and say, "ah, now let's see some real world examples", that might have been a lot easier, and might actually have made question four a little bit easier for me to do, I might have actually been able to give an example for that, because I would have had something to talk me through it.
Pref: [syntactic] I think if I had to use it for any length of time I'd prefer the boxes that you gave me, but this is the sort of thing I'm more familiar with in my background, and so I have to spend less time fiddling around with it. I mean I was starting to... the conditions under which you put asterisks above or below the line, for example, entertained me for some considerable time. [laughs]

- Pref: [Learning Environment] I think would probably be easier to comprehend. But I'm not sure that I'd then be able to extend it into any of the other notation. Erm... I suppose into you diagrams, yes, I could fit that into your diagrams. With a certain amount of effort, translation would be pretty straightforward. But to go to anything like that [indicates syntactic] I wouldn't have thought so. Not without quite a lot of effort.
- Pref: [Learning environment] That one looks simpler. It's less intimidating. I'd say, even coming with the expectation, yes, I ought to be tolerably competent at it, it's still quite intimidating.
- Other: Have you ever come across the game wff-en-proof? [Interviewer says that they have] I used to play that at college with my wife's friends who were all mathematicians and I got murdered. Not particularly because I couldn't understand it when it was there, but it was just generating them.
- Other: I would have liked a few more examples. Having just one example to look at was a problem.

Subject 20

- Studied: Introduction to Logic
- q3: In my reading of the instructional material I had not really understood the section on which this depends. [n.b. stacking & cancelling Modal operators]
- Transfer: The statement and explanation of "each world is only related to itself" I do not understand. That is where my problem begins.
- Sys: I did not understand at all the differences between T, S4 and S5. I had not enough time to really struggle with the differences.
- Proof: Step 4, "construct each new possible world", I do not understand.

Subject 21

- Studied: Logic and Machines
- q3: Didn't really understand the question and what was required. What did you mean use the symbols T, S4 and S5? To do what exactly?
- Ref: $2 \times 3 = 6$ no matter what occurs, therefore 2×3 cannot $= 5$.
- Style: Examples didn't really relate to the questions. Had difficulties understanding questions themselves.

Subject 22

- q3: Once it moved away from example sentences into symbols and equations I became totally lost.
- Pref: Just syntax. There seems to be less chance of ambiguity in interpreting written text than when interpreting a diagram, although a SIMPLE diagram may help to confirm what is in the text but should not be used to introduce a new idea.
- Style: I didn't really understand anything from the section headed "Modal Systems" onwards!

Subject 23

- MOp: Yeah, I think that was alright.
- Sys: I found the difference between... the basic system T alright, I found the difference between S4 and S5 a bit difficult to understand apart from the fact that there's more connections. Maybe that was all it was.
- Proof: I tried to work out, putting into words I could understand, whether p could be valid, and then, what that meant, and then what that meant [pointing to both parts of the biconditional], and trying split it into chunks, and then try to work it through. [Interviewer prompts] Because I was trying to work out if it was valid or invalid. In which case it could be or couldn't be. So I was trying to go through, saying if you can say the whole statement would be invalid, or when that would be the better option, or not.
- Proof: Very difficult.
- Style: I think it's probably not understanding the symbols actually, not being familiar with them, I mean, I understand all the different components, it's more how they related to each other.

- Style: I thought that was quite a nice way of doing it. I would actually have liked another example as well.
- Pref: [Learning Environment] I think I'd prefer that actually.
- Pref: [Learning Environment] Oh god... but you don't need all the squares, do you?
- Pref: I think I'd probably prefer to see it both ways [Syntactic and Learning Environment], which is probably difficult; I think I probably prefer the graphic out of choice, but I think I'd actually like to see the, this sort of way of proving it as well as that.
- Other: I found some of the symbols quite difficult.
- Other: I thought if there'd been another arrow on that end, it would have been easier... but maybe that's just my version of how you can connect things up and down.

Subject 24

- Transfer: I have no background knowledge, and I didn't glean enough from the text to have a go.
- Ref: No, for me, I just looked at it and thought "mathematical concepts", that's what it meant to me, and not symbols. And of course, symbols can mean whatever you like.
- Sys: Erm... I understood everything fairly well until I came to this, actually — "systems of Modal logic". The relationship that seems to be stated here implies that that's is the main relationship that you seem to deal with in the tests and things like that, whereas it refers to other relationships such as, "change one property of an object", or "remove one object". Now that was a sort of a jump that was a bit too far for me to understand. Now, here, you're relating things to the real world, to real world examples, and then there's a jump to abstraction that's a bit too much for me, I think.
- Sys: The differences between them... at a pure abstract level, erm... S... T, the relationships to the next world were sort of one way, weren't they. S4, the relationships could come back, and you could link to another world, whereas S5, all worlds have to be linked to each other. That's all I remember.
- Proofs: Because I didn't understand the true nature of these models, here, these systems, I couldn't relate the truth of them, or not, to the exact models.
- Style: Some sort of relation to real world examples might have helped me, personally.
- Style: I thought that the tree diagrams were fine, if you'd had enough concrete examples at that time to relate the abstract notation, because that's all they were, really, to the real world examples, then I could have understood.
- Pref: [Learning Environment] That would have helped an awful lot, I think. [Interviewer prompts] For one thing, even saying that it's about the way things can relate to one another helps another lot, because to me, even at this level of abstraction, I couldn't even, I find it difficult to think even in those terms, the way things are related.

Subject 25

- Ref: Ah Right... Well, I was thinking about it. I think I was thinking... I was thinking about it in detail, in my answer when I was writing, but I didn't put it, I put just "always 6".
- Pref: [Learning Environment] I think I wasn't drawing any diagrams but I think I was drawing things like this, to see the relationships and so on. So I think that the diagrams, that would be very good.
- Other: I think the problem is maybe because I don't have this background knowledge of basic logic at least, then it's difficult to put this information all together. Perhaps with a little help like ok if you explain to me what these terms mean.

Subject 26

- Studied: Maths (section of Computer Science)
- q3: Yes... My brain melted about then... A lack of understanding: Modal logic is not easy to learn in half an hour!
- Transfer: Although I could see that the statement was true, proving by logic was rather over complicated!
- Proof: The idea of Modal proofs requires more depth to give a real understanding of the subject.
- Style: Type 1 would have been most suitable for me, as the method is rather more linear.

Subject 27

- MOp: The difference in the distinction between red currants and Redcurrants is clear, and black birds and Blackbirds, so I picked that up...
- MOp: All I really got was the diamond was possible and that was it. The square was necessary. It really did not go into explaining those symbols before it went into the implementation of those symbols. And it did not give you... even if it related to a sentence, where the symbol, and, like "dog..."... and then something else means whatever. If those symbols had been implemented into the Blackbirds and the Redcurrants then you could have followed the train of thought, switched on; but it suddenly became very abstract.
- Ref: No. To me that's the only reasonable answer. It's playing with an absolute term. [Interviewer prompts with a name example] Ok, because he could be somebody else.
- Sys: I'm really not clear what you mean, so obviously I have no idea.
- Sys: Up until then it had been trundling along and then... it just goes off the Richter scale, and urgh, I just don't understand what's happening.
- Proof: No, because I was lost by that point. I knew there was no way I was going to be able to answer that, it was just too abstract, and if I was a student studying that I would have had to have gone to look up other material, it was much too abstract.
- Style: It didn't start to define this very clearly, nor did it relate this to something concrete. It did with the first bit of the material, where it starts to talk about black birds in the intervention there. But then it went off into something completely conceptual, with no switching from concept to concrete and back again; there was no relationship between what a person might have been able to relate that to in the outside world and the concept. And it did... when it gave you these expressions, it did not explain every part of the expression; this means this, this means this, this means this, this means this, and put it into something you could understand. The material was very unclear, it was very abstract.
- Style: If there were some relationship to something more concrete... If there's not really a background that's concrete, then, um... it was too abstract. There was none of this switching. So it was difficult to make a symbolic link in your head between something you could link this to, that you have some prior knowledge of.
- Style: Ok, if they could have taken the statements and used some of these symbols with them to try and ease gently into it... ok, you get to about here... ok you're starting to hit back, but once you get here, it's just ok, forget it. You know this stuff's just so... It talks about things being valid, and I start to get lost here, the explanation's very poor, and from here on in, I was really grasping for something I could get a handle on and I couldn't. It introduced the symbols here. For example, this is so convoluted, this sentence, I wasn't sure what it meant. Was this symbol part of the notation or not? I couldn't decide. It doesn't really explain then what it means by world, it doesn't explain what these symbols are, and it's saying this is investigating the truth. Ok, I haven't seen these things before, I haven't seen this notation, what does this mean? What does this mean? Would someone explain this to me, then I might have a chance of understanding the rest of it? But I just looked at this and thought, what? Ok, that means possible, that means necessary, what's this, and what's this? And I thought does this mean that or what? I didn't look this up, so I guess that was my fault.
- Style: I'd like to see something diagrammatical. I'm a picture person. Some of us are textual, some are pictures. I'd like to see something in there which helps those who learn more graphically.
- Pref: [Diagrammatic] I think that by virtue of the fact that you've given me a conceptual explanation of what this is trying to do instead of something concrete to start with, that has locked my understanding. But if I had a concrete example first and then moved from that to this then I think that yes, I could follow the train of thought. But that was why I was asking for a concrete example so that I...
- Pref: [Learning Environment] Ok, I'm automatically drawn to this because this is a picture, and I'll look at the pictures, and see if I can work out what's happening. Ok, it's given me an explanation. Fine. I'm getting a bit closer. Right. Ok. It'd take me a bit more time to sort that, but immediately there's a lot more for me to grab here, and ah yes, there's some representation of what's happening. Yes, the graphics help tremendously. [Interviewer prompts] Just from what I've seen of it, it would appeal to

- me. I feel... I needed much more explanation of what was going on, and the graphical representation for me helps. If you noticed in what I was doing, when I got in trouble, I was drawing things to help me reason.
- Pref: [Learning Environment] I can relate this to some prior experience or understanding I have, because obviously it's a graphical representation I can understand, and that this must fall between here and here, it's a connection I have just made. And it's very important for me to have something pictorial, me personally, I don't know how important it is for other people. But for me pictorial representations are extremely important to get the understanding. And for me, this gives me a connection to other things I can relate to.
- Pref: It's to do with the fact that you can actually see. It's almost tactile in a way, although it's not tactile... virtually tactile! [laughs] [Diagrammatic] This certainly doesn't aid my understanding. It's not relevant to me, because it has no connection with anything that I've done before, so it's not a picture as far as I'm concerned, it's words with boxes round them. It's not a picture. A picture tells you something. So I would argue that this is not a pictorial representation, because it's not meaningful, there's no connection between the representation and the meaning.
- Pref: [Learning Environment] This is something most people could identify with. They could see a ball, to me, I imagine this as a ball, I know what it is and there's no problem there. This is a graph, it's got boxes. I've seen this before.
- Other: I started to feel that that was testing my knowledge of geography, and I became rather apprehensive because my geography's not very good.
- Other: If you're testing logic I'm just wondering were you deliberately trying to test something they knew nothing about to see if they can work it out or do you want to test something that they've got concrete of experience of?
- Other: It depends on what level of understanding you're at. I'm at the stage of knowledge acquisition; I think that this [Diagrammatic] is aimed for someone who can synthesize. I think it's a higher order skill to aid your understanding. So therefore I feel that for those that are just looking at this to acquire knowledge, you're asking them to acquire a diagramming system which means nothing to them. But if they're synthesizing, they will have gone through acquiring the knowledge, and the understanding, and you know, being able to do all the other things. So that would mean something to them. I think that's a higher-order level of representation.

Subject 28

- Ref: I just thought that in case there were people in Louisiana were interested in the bridge they might call it the same name.
- Sys: The only thing I understood was that the common systems are T, S4 and S5.
- Style: Um... When I was reading through the other paper, although it was written, it was too lengthy, and it was difficult to understand. [Interviewer prompts for the problem] I think the symbols. Or maybe I'm not good with the symbols, so maybe that's the reason.
- Pref: [Syntactic] This might be a bit easier than the... than the boxes and symbols. [Interviewer prompts] Maybe I'm just good with words and not with symbols.
- Pref: [Learning Environment] Yeah, it would be easier this way. [Interviewer prompts] I think that I would prefer this one better than the symbols. [Interviewer prompts] This is easier to understand, I think. By looking at it you can see it, but with symbols it's a bit difficult.
- Other: Because not very much time with all those things to understand it fully. I think you have to read it many times to understand. I haven't done anything like this before, that was the other reason.

Subject 29

- Studied: Physics GCSE (electronics)
- q3: Felt that there was too much information to absorb - maybe doing some examples as I tried to learn the material would've helped. I looked at these questions and blanked.
- Style: I just felt swamped with info. Maybe a test after each idea would have helped me understand. Wasn't sure of the overall objective behind the questions asked. I got lost when arrows linking worlds were brought in, especially when the 'bypassing arrows' occurred!

Subject 30

- Studied: Mathematics degree
q3: I had to translate it into words in my head and then got more confused!
Pref: I would have liked diagrams but didn't find these very helpful as I didn't understand the notation. I would have liked practice exercises to build up my confidence.
Style: Too many new ideas / new notation to take in in one reading. I would need more time / practice.

Subject 31

- q3: I don't know how to deal with these things.
MOp: I don't understand what these symbols do to p or q.
MOp: Why do we use them? To see what happen when p and q together in different conditions and see the result?
Ref: No, I didn't improve on that question at all.
Style: The language is not very familiar, things like reflexive, I have no idea what it means in this context.
Style: Trees and boxes and things are alright, but I think they didn't help me to understand what is the real thing going on.
Pref: [On Learning Environment condition] This looks better, but as I say, I should be able to understand them. But this looks easier. [Interviewer prompts] It's... You can move the ball, and then make a statement, and at the same time see what happens to other statements, so you can, you know, relate them to each other better. Here, I was wondering if the q is always the same, and there are two diamonds before it, etc. Maybe I had difficulty in conceptualising it here, but if I see the parts then I will know this means diamond diamond q p whatever. I don't know. If it is the thing you want to teach, otherwise...
Other: I didn't understand... I mean, I needed time, I felt. Maybe with time I could learn the principles, can understand there are some principles, if I had time.

Learning Environment condition

Subject 32

- MOp: They're so unintuitive, really. It's not like... you're used to certain symbols, as suggesting transformation, but the diamond and the square don't in themselves suggest anything, until when you first come across them you're trained to sort of slot them into boxes in your head, and that's quite awkward, I think.
Transfer: So all I would have needed to do was draw a box and I'd have got full marks. [laughs] [Interview prompts for an example] Basically you're talking about any situation that can't change, over which you've don't have any control, so your plate's empty, you want more, but you can't move to a different world, so your plate stays empty.
Ref: I remember it being mentioned, but I was taking the thing more at face value, I wasn't looking for tricks.
Sys: Not... not really. I started to think that even given what I've just read, I couldn't honestly define what a Modal system is, and as soon as I read it, I thought, I'm just feeling out of my depth. I did try, I mean, I thought I'd read through it anyway, see if I can, but I mean, I couldn't really.
Style: Right, I'm a bit unsure to put it in, erm, algebraically.
Style: Um... ok; fairly clear, I think. I'd mention that the earlier that there was talk of levels, and that was confusing, as the tree diagram had to be represented in several layers, but I think, myself, I can sort of sympathise with that sort of representation.
Pref: [Diagrammatic] Lovely. I'd have thought, get me something else, really! Jams jars and spoons, or something. [Syntactic] Er... well again, they're not... as someone who has not come across this sort of thing before, they're just not intuitive, and you just have to read a lot of them before they started to make sense, before you get over the hurdle of always having to interpret exactly what the sign meant, it's not like coming across $3 + 5$, where all your background, means that you know already what 3 and 5 mean. If you then have to go off and hunt, like in a different language or

alphabet system, what those symbols mean, it complicates things. [Later] If it was done fairly closely with personal instruction with a tutor or somebody talking you through it, then it might make sense, but if you're supposed to be able to open a book and follow it, I'd say not really. You'd have to be bloody determined!

Subject 33

- MOp: I'd like that to be slightly more clear. I didn't find that very clear.
- Sys: Thought I read the material you gave me, I still am not sure that reading all this text made clear to me what... how I could move from one Modal logic system to the other, how I could do this simplification and things, so the symbols in themselves... I was slightly confused, for instance, between the different systems, so I could not really tell what the difference was, so this meant that I could not... I wasn't able to translate between the logic and the symbols very well.
- Sys: I'm clear about T and S4, but seeing S5 kind of confused.
- Proof: Um... I'm not really sure.
- Style: I found that very clear, very interesting. Yeah that was the most interesting part of the reading material. You know because it's quite compact, you go back to something compact, which will make... you can see it done, instead of [indistinct] with your mind.
- Style: I like, I like diagrams, because it's the only relation of this abstract mathematical stuff to something, you know, touchable, something which is real world information. So if that's the option, I'd rather go with this one.
- Style: I guess you are using these diagrams with the brick walls and everything to make logic more accessible to students, but thought it's quite nice and I like it, sometimes there's no direct correspondence between the diagrams and the mathematical symbols and the mathematical relationships. So that's where I was lost. When I went back to the post-test, I had a problem with getting back to the logic, and this led me to revert back to my initial notions about the meaning of the sentence, not related to what I had read in between.
- Pref: [Diagrammatic] I think that's nice, probably, because it gives an emphasis on the relationships which may be lost in this, or in the text.
- Pref: [Syntactic] I prefer the first one [Diagrammatic], if I had the choice. I was not really clear, I was quite confused, how these nice diagrams translate into something like that.
- Other: I find all the symbols slightly frustrating, you know, having to remember all the... and, I don't know, probably because I am not a native speaker, I try and stick to words, you know, try to construct meaningfully words, and then when I try and translate this in mathematical terms, I find this slightly confusing.

Subject 34

- Studied: Introduction to Logic
- q3: If you say out loud what it says symbol wise, symbol by symbol, it doesn't make sense.
- Transfer: To begin with I didn't understand the new system so struggled.
- Sys: The fact that they had different rules and remembering those rules.
- Pref: I think a mixture of syntax, diagrams, and simple examples with diagrams would suit me best. Particularly the diagrammatic examples.

Subject 35

- q3: I really couldn't cotton onto the symbols.
- Transfer: Couldn't understand the question.
- Style: In a very pressured schedule found this difficult to engage with; there was nothing in it for me (even at an intellectual level) & it made me feel thick! I found the language, density and level of discussion inaccessible and uninspiring.

Subject 36

- Studied: Logic & Machines
- q3: I couldn't understand the basic principles.

Subject 37

- Studied: Logic and Machines degree module
q3: I didn't really understand it
Transfer: As with q3 on the tests I didn't understand them.
Ref: Because the proposition wasn't true and the symbols are not going to change meaning it was difficult.
Sys: They were complicated concepts to grasp.
Proof: They were complicated concepts to grasp.
Style: I think more diagrams with step by step explanations would have helped.

Subject 38

[Note: Full transcript provided in Appendix 14]

Subject 39

[Note: Interview lost when audio tape snapped]

Subject 40

- Studied: Prolog
q3: These symbolic questions were problematic: not because the instructional section was unclear or badly written, but because it's a bit much to take in all at once. Given several readings of the materials, and step-by-step instruction (with example answers) I could approach question 3 with more confidence.
Transfer: I think my first-order logic training helped here. The instructional booklet presented the concepts of S4 and S5 well, but the answer still demanded a symbolic representation, which I'm a bit fuzzy on.
Ref: No, no problems. It merely assumes the possibility / impossibility of other worlds, depending on the meanings of the symbols used and the rules applied to all worlds. I also assumed, generally, that S4 / S5 applied in these questions.
Style: 'Syntax only' demands some imagination to be fully useful, and can be misconstrued. 'Diagrams only' can illustrate, but are useless if the diagrams don't teach things which are useful in related situations. Diagrams with simple examples are good, although it helps if the diagrams are broken down into stages, with accompanying syntax or symbolic representation, to illustrate how those diagrams would look in a different format.

Subject 41

- q3: Don't know what symbols mean - and glossary irrelevant to my 'significance' gap.
Ref: Don't care. Too steep a (memory) learning hill to climb.
Ref: I wondered whether it referred to 'knowledge' out there I don't have re maths, symbols & proofs - their current truth status!
Pref: On types 1 & 2: You need a designer! This is conceptually / visually awful and not interactive. Unreadable and unlookable at - shows no sense of shifting eye and conceptual problems!
Style: Like other people's diagrams — it/they reveals other people's thought processes — but doesn't make them mine. This is *alien* thinking and describing. 'Exposition' won't bridge the gap between us — nor will representation — I need an EXPERIENCE (a doing).
Style: Haven't been supported to understand - the gap between us (your knowledge and your consciousness of the problems) and mine has not been addressed. This all comes from you. It is one-way teaching and thus "don't work"! It was hopeless for me. Needs to be built up more simply (in stages) and attractively (differentiating in colours) and related to a real situation we all know.

Subject 42

- Studied: Philosophy
Transfer: A lot depends on how wide a range of interpretation is allowed for "necessity".
Ref: Could 5 actually mean what we mean by 6?

- Pref: Diagrams with examples are easiest because misconceptions are usually cleared up quickly.
- Style: I'm unsure about the extent that the semantics of the accessibility relation really "captures" the meaning of the Modal operators.

Subject 43

- Sys: Yes; starting to.
- Proofs: Um... constructing Modal proofs. Um... no; no.
- Style: Well, what I need, and what I'm not sure that it's possible to have, I'd need a kind of going backwards from this to a kind of natural language equivalent, to a kind of real world equivalent. But I'm not sure that it's possible in something which is so self-contained.
- Pref: [Diagrammatic] Unnghh... ok, my initial impression is... that that is... that would just completely freak me out. But you know I haven't read the whole thing through, bit by bit, so maybe if I did, I would feel a bit better about it. But my initial impression on looking at it, is that I would look at this, and initially just be that much happier with little boxes doing things that I understand. [Syntactic] Umm... I mean... somebody like me who doesn't have a strong mathematical background... algebra is... algebraic terms... no. You need to know the rules and regulations there, whereas at least with this you can say what the rules and regulations are, and then you can do a physical demonstration of it. There's an awful lot you need to keep in your head. With this sort of formulation you also need to keep in your head the algebraic formulations which you probably haven't done, well, which I haven't done for a very long time. Which doubly compounds what you're having to cope with.
- Other: Um... I think... I felt ok about it. I mean I felt that it's an horrendous subject, and I could definitely feel it slipping away towards the end. Erm... But I felt that it was a fairly good grounding, and that if I did it again several times then I'd know what I was on about.
- Other: I think actually, a lot of logic I actually find quite good fun if I know what I'm doing. Um... but with the modal stuff, I still haven't quite extended it far enough so that I see how it applies to the real world, if it ever does, and so I'm not confident enough with it yet to say that I actually enjoy using it.
- Other: I'm very aware, because I'm involved in sciencey areas, I'm very aware that I'm a linguistic person, and I find it difficult to look at boxes and diagrams and to get sense from those.

Subject 44

- q3: I did those simply by learning that truth table, learning how the transformations went through the symbols, so I can see here, for example, that S4, the first symbol in the series is transformed, and in S5, the last symbol in the series is transformed, and T seemed to me, that there was no transformation here, so I didn't include that. So I'm not understanding that at all, I'm simply doing it syntactically.
- MOp: Well, that's difficult, because I can see they have two... I mean, you ask me if I understand them, but really what you mean is can I use them, because although I understand them in natural language terms, I probably couldn't necessarily guarantee to make a valid equation using them, because they have different meanings from the natural language terms.
- Sys: I don't really understand them, no.
- Style: Erm... what, you mean as diagrams, basically? No, I thought that was quite good. I thought the boxes are quite, erm... clear.
- Style: Well, I though the grid was interesting, was slightly distracting, which wasn't referred to. For example, it was given at the beginning, for example the circle can't cross a wall, but it didn't play any part in the exercise or the understanding of the three systems, as far as I can see. I can see how that'd be useful in a further development giving more complex examples, but in terms of the examples we were dealing with... [Interviewer prompts] They were alright, it's just that the wall added a further set of "what ifs..." in my mind that weren't relevant to what I was dealing with.
- Pref: [Of the diagrammatic condition] It's as clear as mud. [laughs] I'm sure if I'd started on chapter one, and worked through... well, I mean I did, didn't I. I did read it

through up to this point. But where's the point where it breaks away from there? [Interview shows him] Right, so that's just where you went into... Well, I mean the boxes took me a lot further than this does. I mean, even now having read the boxes, I'd now have to go back to the start again. I mean, my view is, even though I ended up not being able to reproduce what I thought I understood, the boxes still made me feel I was understanding it, whereas this just tell me I don't.

Other: Well, you can't really, can you, because natural language and the way you think interferes with understanding how the systems work.

Other: I think I did better with the pre-test than with the post-test.

Subject 45

q3: Just couldn't get to grips with the symbols at all - I didn't understand the abstract theoretical side. It made sense when talking about actual concepts, but this was too abstract - I was lost. Symbols etc. needed more explanation - with examples, e.g. from real world (or defined world).

Transfer: I was completely lost. As I said, I didn't get the symbols in the first place. This was way over my head.

Ref: Difficult to justify when all you can say is "that's just how it is". Found it quite easy to state necessarily true or false though.

Pref: Diagrams with examples - I like examples, as it means you can see how things are working.

Style: I think there's too much of a leap in concepts - I can cope with it in words & examples, but not in symbols & theory.

Other: Not sure at all about this bit - could be necessarily false because e.g. there'll always be babies etc. who aren't yet uncles. I don't know whether or how much you're meant to look at real life.

Subject 46

Studied: Logic & Machines (Maths) and Artificial Intelligence

Transfer: Not too sure what we were supposed to produce. More info could have been better?

Ref: It was a straight yes/no, which implied a confusion of mathematic signatures! You either had a correct 'single' answer, and multiple incorrect answers possible.

Sys: Easy!

Pref: They were very well represented, but could be quite confusing to a person not experienced in AI or Comp. Sci. Very good use of syntax and examples. Symbols could confuse! Very good information given.

Style: I just got confused with all those [lists string of operators] all thrown up at once.

Subject 47

q3: I think question 3 was the one I saw some improvement, like, oh now I know what to do with these operators, because that's what puzzled me with question 3 of the pre-test. It's that it was too complicated to go on thinking "must be that it could be that it must be that it could be" so there must be a way of dealing with this without thinking, and that's what I was happy to learn in the thing I read, so I went straight there because I want to know to collapse the loops, or...

MOp: That's what I felt more comfortable with: learning how to play with those operators.

Sys: In S5, it's only the last one that counts. And in S4, you can sort of pull up the repeated ones, either something or a pattern. But I am not sure... so... I have got how to operate them, but I am not sure I able to reproduce the explanations in my own words. So I can play with operators, still have some idea, but if... When I go back, when I am seeing the trees, I think I can understand that, but if I had to re-explain that I wouldn't be able to do it.

Sys: Yeah, so I'm not sure I would be able... no I'm sure I wouldn't be able to explain these three ones, but, um... so I'm not sure I learnt what this is about.

Style: I thought that that was much easier than I thought, especially when I was trying to answer question four, and you were saying, "which world..." Then I remembered that worlds were represented in that way, so ok, then I can answer using that world while, when I was trying to answer question 4, especially in the pre-test, I was thinking of

any possible world, like oh yeah, there's a place where London Bridge could be in Louisiana, you know but, it's easier to think of circles, and the walls, yes, and when you're talking about above or below B, then it's ok, if there is a wall, then it cannot possibly be that. I think if I had thought more of the representation of worlds through the text, things would have been clearer.

Pref: [Diagrammatic] No, I get confused in this one, but I dunno, I saw that one first, so...

Pref: [Syntactic] No way. [laughs]

Other: My difficulty is, I say it, but I cannot draw it, in a way that, I feel difficulty to explain it with this way of proving it, because if I say, well, so if, if p, then, then, then it must be p because there is no other way you can go.

Other: I read that too quickly. I think I would have liked to have read it again and again to understand it.

Other: Sometimes I was reading and I was just getting a bit confused, I was saying, "ok, I'll understand that later, it's ok," when I hadn't actually understood.

Other: One thing which could have helped me, was if I had exercises along... along the thing there I would be getting used to it. Because at least for me, if I go on reading, I say "ok, ok, I'm getting it, I'm getting it — oh, I just want to see this picture..." I felt very much unsure about the things I had learnt.

Other: If I had more time to read it through, then now that I have had the first read, then I would be able to understand it better.

Appendix 4: Interview transcript for subject 38, first study

[looking at Exercises 1]

SUBJECT: Ok, but that's... they're merely hyp... they're hypothetical. I can't see why that one... Oh dear. Because it says here, $2 \times 3 = 6$ no matter what happens in the world. But you could imagine a world in which $2 \times 3 =$ something else.

INTERVIEWER: Like...?

SUBJECT: Well, the multiplication symbol may stand for something else.

INTERVIEWER: And what does the proviso say about that?

SUBJECT: Sorry! Right... Sorry. Yeah; but... they're meanings. But so is this. The symbols change their meaning but our interpretation of Mickey Mouse and Michael Jackson could change.

INTERVIEWER: You don't see where the definition's from?

SUBJECT: No.

INTERVIEWER: Right.

SUBJECT: It's... oh. It doesn't matter. I'm probably getting myself tied up in knots.

INTERVIEWER: No, this is one of the problems which comes out of this. You've got to specify what you're talking about before using Modal logic.

SUBJECT: Ah...

INTERVIEWER: And one of the things that you can talk about... Normally, symbols are assumed to be their names, if you see what I mean. They are symbols with a meaning attached, but it's assumed that they're directly representing it. But names, it's treated like a name, and the name has a reference, which is some object, which is a real-world object. That's why there's a difference.

SUBJECT: So that could be x and y . But that symbol is assumed to be that symbol, whether we lived on the moon or whatever. Right, ok.

INTERVIEWER: Does that clear it up?

SUBJECT: Yeah, it does.

[after]

SUBJECT: Possible worlds really annoyed me.

INTERVIEWER: Why?

SUBJECT: It's like it's... um... it's like these forms of logic. I don't know... sort of re-inventing things, making assumptions. Like the idea of a possible world. I'm probably going off track here. But if you bring in the idea of a possible world, you could be here forever, saying, "well, in a possible world, we can assume... this and this and this." And I can't... I can see...

INTERVIEWER: You can't see how these possible worlds might be connected in some way?

SUBJECT: I suppose what I'm very confused about is the use of this.

INTERVIEWER: Ok, so let me give you a couple of examples. Things like... [background; outline area, contingency plans, etc]

SUBJECT: And when could that be useful?

INTERVIEWER: Say you're planning a picnic, and you want to know whether you should take an umbrella or not. Well, will it rain or not? Well, it could do either so I'd better take an umbrella just in case.

SUBJECT: Right, ok, ok.

INTERVIEWER: But it's used more in programming and stuff. It's also used in things like linguistic philosophy where people argue about the meaning of sentences with ambiguity, and you're trying to work out is there certain common ground between these arguments, irrespective of their initial position? And it's used to define what things like truth means [explains a bit].

SUBJECT: So they're basically getting down to the bones of it.

INTERVIEWER: Yes. Does that help at all?

SUBJECT: Yes, it does a bit. I think I'm... I think I was introduced to logic in a way that was very tricky, and it... yeah, it was just a bit odd.

INTERVIEWER: Which was...?

SUBJECT: Which was on this knowledge representation course and we were introduced to these two... two that I mentioned. And we just mucked about with truth tables, and you know, symbols on a table, and it just drove me up the wall.

And I... that's all. I just think it's so removed, it's like playing games, it just irritates me. Probably because I don't understand it. But then I think, because I'm intelligent, I should be able to understand it. Perhaps it's how its presented.

INTERVIEWER: So having explained this idea to you, do you understand the idea about [references]

SUBJECT: The bit we talked about? Yeah, I think I have.

INTERVIEWER: Why didn't you answer question 3 on these tests?

SUBJECT: Because if I do things I like to get them all right.

INTERVIEWER: [laughs] So you didn't do it just in case...?

SUBJECT: No, I'll tell you. I don't understand what it does if... that's double. You see, I could look up that; I didn't really have a... first of all, having read the thing, I didn't understand what the three different systems were about. It was like, having three different systems, if I remember it, was like creating more assumptions. Different assumptions. That's what I understood, how I remember it. Um... See, there was, in the bit of the learning material you gave us, there were three Modal systems, and you could predict different things in different ones.

INTERVIEWER: Like?

SUBJECT: Like I think as you went to S5, I think they were introduced as T, S4 and S5, I think you could predict more, you could predict things like "What happens in the future?" more.

INTERVIEWER: As opposed to...?

SUBJECT: Just what's going to happen immediately next, as in T. But it was so abstract, you see.

INTERVIEWER: Right; ok. Can you think of any way in which it could be made less abstract, and in which it could be improved?

SUBJECT: I'm not saying I can't understand abstract things, but when it's a new area...

INTERVIEWER: Oh, yea, I'm not...

SUBJECT: No no no, I'm just trying to think. Or perhaps I can't understand them [laughs]. What, any... I'll just tell you about this. Well, I could look up that, and then I could... there were some rules here, something which reminded me of predicate... something I'd come across. But there were two of them. So I didn't know what to do. So I was stuck. And there were two here.

INTERVIEWER: And there's lots on the next one.

SUBJECT: Yeah.

INTERVIEWER: And you didn't get that out of the instructions?

SUBJECT: No, not at all. How could it be made less abstract? I don't know actually. Because when it was talking for example about Micky Mouse and, erm, you... usually, you think, to make something less abstract you use concrete examples, you use familiar examples, so when it was talking about Micky Mouse and Michael Jackson, my brain still got in a twist. I don't think it makes much difference at all. [laughs] Total twist. You could have been talking about A and B, it would still have got in a twist.

INTERVIEWER: For any reason?

SUBJECT: I just don't know. Erm... let me think. I think what I got confused about is this idea that... well, for example, I don't know what the answer is to that. But... oh, I don't know. Where you're talking about beliefs for example anyone could believe what they like; I might believe all sorts of things.

INTERVIEWER: Yes. So is it possible that you could believe that $3 \times 2 = 5$.

SUBJECT: Yes.

INTERVIEWER: Ok, that's right.

SUBJECT: Ok, is that... I don't know.

INTERVIEWER: Don't know what.

SUBJECT: I don't know why I'm getting it... You said you wanted to know why I got in a twist about things, even if concrete examples are used, and I don't know why, but there's a lot of confusing ideas, I think. You're making a lot of... it's the type of logics you describe, they make a lot of assumptions, and I suppose you'd have to get used to those assumptions and do some exercises and use them before you learn the next form of logic. Do you see what I mean?

INTERVIEWER: Could you give me an example?

SUBJECT: Well, first of all, your first order logic was briefly introduced. Now, it would be nice I suppose if I was to learn first order logic. I could learn all the assumptions, the rules, and conditions it makes, and all the symbols, and then you can muck about with it, do exercises in it. And erm... and then you said Modal logic was built on, derived from first order logic, 'cause first order logic failed in a number of ways, limitations, it can't do certain things, so along comes Modal logic. And having first got the hang of first order logic, then you could move into Modal logic.

INTERVIEWER: Yes, but you see I can't teach first order logic as part of the study. [explains teaching level of Modal logic; i.e. after a first course in logic.]

SUBJECT: No I don't think it's just a problem with that [i.e. FOL]; no, no. Because most of the sheet covers Modal logic. I'm sort of thinking in the real world what would I do? That's what I'm thinking.

INTERVIEWER: Do with what?

SUBJECT: With learning Modal logic. I sort of would assume that once you've learnt one form you'd sort of find it easier to pick up another. That's what I'm saying.

INTERVIEWER: Do you want to have another look over this, so that you can point out where the problems were?

SUBJECT: What, in reading it, as a, as a, learner? Right.
One thing that threw me initially was this idea of contingencies and I always think of contingencies as things having a relationship.

INTERVIEWER: Like?

SUBJECT: Things being contingent on something.

INTERVIEWER: Right.

SUBJECT: Because it has that meaning as well, outside of logic. And so that threw me, but luckily, you put in brackets what you meant by contingency in this, but that just initially threw me.

INTERVIEWER: But you've got the idea of it now? This one?

SUBJECT: Yeah, and I did get that, roughly. Possible worlds... That didn't seem too tricky. I mean...

INTERVIEWER: Did you understand accessibility relations?

SUBJECT: No.

INTERVIEWER: Right. Could you try and explain them to me, and we'll see what it is about them you don't understand.

SUBJECT: Well, I don't think I really understand what a Modal system was. I did talk about that earlier.

INTERVIEWER: Ok, forget that. Let's just talk about accessibility relations, and we'll go one step at a time.

SUBJECT: Can I use this? I don't remember at all. I just remember not understanding what an accesssab... an accessibility relation was. Oh; it's up here. See? I didn't even remember that!

INTERVIEWER: [Laughs] She hasn't even read it!

SUBJECT: I did read it! Yes; I thought that was a big problem with possible worlds [i.e. infinite amount] [draws a note] This is an infinite number... I'm very visual... of possible worlds. And they could relate in a number of ways. Oh; right. Yeah, I get that. All this... all that Modal logic does, is that it imposes a structure on this set of worlds, by saying this is linked to that, or... it could be linked in another way.

INTERVIEWER: Like?

SUBJECT: Well, like it says here, "wait five minutes."
It goes from here to give some examples of possible accessibility relations. How do you know when to stop here? Is there a finite set of accessibility relations?

INTERVIEWER: Well, what do you understand by it.

SUBJECT: I would... because here it says... I would say that there was a finite set.

INTERVIEWER: Because?

SUBJECT: Erm... Because there is. Because you have all these possible worlds... Well, each world can... I mean these examples are sort of taking me down a track anyway. Well, like there — do nothing. The world could do nothing. Or it could bump into that one. So it could change.

INTERVIEWER: What do you mean by bump into?

SUBJECT: [Laughs] I don't know, I'm using these as examples and it's throwing me.
INTERVIEWER: Ok; think of your own one.
SUBJECT: There, put that away. So I'll just go from here. I don't understand it. If the possible worlds could be infinite anyway. And Modal logic seeks to impose structure on that. And it does that by saying that... I'm probably repeating myself. And it does that by saying they're linked in... I don't know.

INTERVIEWER: By accessibility relations?
SUBJECT: Yeah. Which are... I presume, how the objects... I'm suddenly thinking of how objects behave, but I don't know why I'm thinking that. I think I'm thinking of your grid. You know, the grid and things, how they move about. I would have thought that otherwise.
You see, a possible world is not an object.

INTERVIEWER: What is it?
SUBJECT: A possible world could be made up of lots of objects. Ah.
INTERVIEWER: But you're treating it as one thing. What you're doing is taking a set of assumptions, like the sky is blue, the grass is green, and that set, doesn't matter how many things are in it or not, will be treated as one object.

SUBJECT: Oh, right. Right. I don't know.
INTERVIEWER: You don't know what.
SUBJECT: I don't know what an accessibility relationship... I don't see why that imposes structure on it. On the objects. That are actually possible worlds.

INTERVIEWER: Say you've got this infinite set of all the different things that could happen.
SUBJECT: Ok.
INTERVIEWER: Right. Well, this is no good, you can't do anything with this, it just goes everywhere.

SUBJECT: Right.
INTERVIEWER: So you say something like well, ok, what am I trying to do here. Let's look at what will happen in the future. So you want to look at all the possible worlds you can get to in the future. So how can you get to them? You can wait one minute. You can wait two minutes. You can wait three minutes. You can wait a second. And you can wait a second lots of times. So you take lots of little steps into the future instead of one big one. And each of those will get you from wherever you started to some number of possible worlds in the future.

SUBJECT: Right, ok. But that brings in the idea of a time scale. I didn't think that was relevant.
INTERVIEWER: Well, you can bring it in as an accessibility relation.
SUBJECT: Ah! That's just one accessibility relation! Time! Ah...
INTERVIEWER: Yes. Another one might be, "add three", if you've got a world where what you're doing is... say you're looking at how many rabbits are in a warren, and you'll look again once there are three more. You can come up with almost anything. What you're looking at is, here's the starting conditions, here's some way of changing that that we're interested in looking at; ok, given this way of changing it, where can we get to?

SUBJECT: Say that again?
INTERVIEWER: Ok, you've got some set of initial conditions, doesn't matter what they are some situation that you've got described, and you say, ok, we want to look at how this can change, in this particular way. Ok, well what situations could we get to if we allow it to change like this? So we could have a situation like, let's go for the picnic example again. [goes over example again]

SUBJECT: There's millions! There's loads and loads. You could go on forever.
INTERVIEWER: Yes, but the more restrictive a relation you impose on it, the fewer you have to deal with.

SUBJECT: Impose on what?
INTERVIEWER: Ok, you've got this great nebulous set of possibilities, and you've got no hope of making sense of it all unless you put your blinkers on and concentrate on one little part of it. And the blinkers you put on are an accessibility relation like "wait until noon tomorrow and see what happens." So you're going to discount almost all of it and concentrate on some part of that.

SUBJECT: I don't know if I do understand, actually. Why I'm confused, what I think what, I'm finding it hard to understand accessibility relationships, because

I'm finding it hard to understand what they mean by a possible world. Because I understand — I thought I understood — and I know you've explained to me about how in a sense they're treated as objects. I just view a possible world as so rich I can't break it down.

INTERVIEWER: Ok, but that's why I've introduced these grids, because they're not very rich.

SUBJECT: Right.

INTERVIEWER: So if for example we take one grid world...

SUBJECT: I mean, I could probably understand it more for a grid world.

INTERVIEWER: Ok, well, do you want to try explaining it in terms of this.

SUBJECT: Accessibility relation?

INTERVIEWER: Yes.

SUBJECT: Can I just look at your examples of accessibility relations?

INTERVIEWER: Yup.

SUBJECT: Oh, I got confused here.

INTERVIEWER: Well, don't let's get to that bit yet.

SUBJECT: Well, what shall I describe in my grid world?

INTERVIEWER: Um... Whatever you want. You don't have to have it as big a grid as that if you want to keep it simple. Just do it like a noughts and crosses grid.

SUBJECT: Can't even do a noughts and crosses now I'm thinking too hard. Like that?

INTERVIEWER: Yes.

SUBJECT: Ok, this is not a possible world.

INTERVIEWER: Why not?

SUBJECT: Ok, this is a possible world.

INTERVIEWER: Why?

SUBJECT: We can say that this is a possible world.

INTERVIEWER: Ok, that's good enough.

SUBJECT: Right, so that's a possible world, but there might be other ones.

INTERVIEWER: Ok. So, what are we going to put on this possible world? Let's have a complete situation, let's define it completely, because otherwise it'll be ambiguous.

SUBJECT: So shall we put objects in this possible world?

INTERVIEWER: You can do whatever you like with it.

SUBJECT: [laughs]

INTERVIEWER: So, what's that?

SUBJECT: That's a circle, because I've got a bias here. And we've also got... no, I don't want to have it noughts and crosses. We've got a circle, and a triangle... can I have one more?

INTERVIEWER: You can have whatever you like!

SUBJECT: And a diamond.

INTERVIEWER: So you've got three shapes and your possible world. Ok, so what can your shapes do?

SUBJECT: They can move.

INTERVIEWER: What, all of them?

SUBJECT: Yep. They can all move.

INTERVIEWER: How far can they move, and in which directions?

SUBJECT: Well, each one... [laughs] can move... I'll look at the constraints of the space in a minute, I'll just say what the objects can do, first. There's going to be constraints in the possible world. Is that allowed, too?

INTERVIEWER: You can allow whatever you like! This is you making up a world.

SUBJECT: Oh, right. So each object can move up, down, left and right. This is very simple.

INTERVIEWER: How far can they move?

SUBJECT: Oh, just one. One box.

INTERVIEWER: Right.

SUBJECT: Now there's already a constraint, because if they're right by the edge, they can't move out. That's fair enough.

INTERVIEWER: And you can't add any in, or take any away?

SUBJECT: Nope. They have to stay there. Is that describing accessibility relations?

INTERVIEWER: Yes.

SUBJECT: What I've just said?

INTERVIEWER: That is an accessibility relation [points out set of notes about movement constraints], this is a possible world. So you've just come up with your own one.

SUBJECT: Not accessibility relations? Are there lots of them, or just one?

INTERVIEWER: That's an accessibility relation.

SUBJECT: All those sort of things I've just...

INTERVIEWER: If you had a different one, say you allowed it to move diagonally as well, that would be a different accessibility relation. It's the conditions that you're imposing on that world to say where you can get to from it.

SUBJECT: Right, I think I get that. I think so.

INTERVIEWER: Well, what's the next step on this? Well, you've got this world, where can you get to with it?

SUBJECT: You know you said possible worlds could be objects? Well, could they [points at diagram] be possible worlds with objects in?

INTERVIEWER: If you really wanted to, but it would be horrendously confusing.

SUBJECT: No, I don't want to, I can't see how they can be both, you see.

INTERVIEWER: Ok, let's call it a situation instead of an object. Does that help?

SUBJECT: What, this?

INTERVIEWER: Let's say a possible world is a situation.

SUBJECT: Yeah, that's better. Now what?

INTERVIEWER: Well, let's see where you can get to from that. How many of these; how many new possible worlds can you get to?

SUBJECT: Loads!

INTERVIEWER: How many? Don't say loads!

SUBJECT: I don't know!

INTERVIEWER: Well, have a look. Oh, that's the other thing. Does one of these have to move, or can you move none of them?

SUBJECT: No, they have to move.

INTERVIEWER: At least one has to move.

SUBJECT: Yes. Well, there's loads, because one could move... did I say they all can move?

INTERVIEWER: No, you said only one can move.

SUBJECT: Oh yeah. Well, either that could move, or that could move, or that could move. So that's three.

INTERVIEWER: How many places can each of those move.

SUBJECT: One. It can only move one.

INTERVIEWER: Yes, but to how many new places can each of them move? This one could move there or there as well, couldn't it?

SUBJECT: Oh no, I was only seeing these as the boxes.

INTERVIEWER: So why are you ignoring these ones?

SUBJECT: Because I was focusing in on the noughts and crosses.

INTERVIEWER: I don't follow that. Alright; alright, let's scribble out those, let's scribble on them because they don't exist. It's just so I know what's right.

SUBJECT: Sorry; sorry; still, that can hold.

INTERVIEWER: That's fine. So now you've got three of these places that you can get to.

SUBJECT: Yes. That can go there, that can go there, or that can go there.

INTERVIEWER: Ok, then draw those.

SUBJECT: So you've got three possible... Oh; I'm drawing it in a box, but it's not a box. Sorry.

INTERVIEWER: It's like a cross shape. That's fine.

SUBJECT: It's on a piece of paper, that's all. So this... it could be that that happens, in which case it'll look like that, or... you don't need me to spell this out.

INTERVIEWER: Do it anyway.

SUBJECT: ...or that could happen, the circle moves to the centre, or the diamond moves. Yes?

INTERVIEWER: Right. Ok. So what can we say about this system?

SUBJECT: I don't know.

INTERVIEWER: Well, let's just think of anything. Um... let's think of how these shapes are related to each other, because that's an easy one, you can just read that off. Erm... can the circle be below the diamond?

SUBJECT: Ah right, you can start predicting... This is when to do with prediction, isn't it? Well... it could be in the future.

INTERVIEWER: Could it be at this stage?

SUBJECT: No, not at this stage. Well, I haven't checked, but I don't think so.

INTERVIEWER: Well, is it in any of these?

SUBJECT: No.

INTERVIEWER: Can the diamond be below the circle?

SUBJECT: Yes.

INTERVIEWER: Right. Does the diamond have to be below the circle?

SUBJECT: Well, I'll have to look.

INTERVIEWER: That's alright.

SUBJECT: Yes, it does, doesn't it.

INTERVIEWER: What if you allowed more than one movement? Would the diamond have to be below the circle?

SUBJECT: I think it would actually, because the circle moves...

INTERVIEWER: What if you moved that to there, and that, and that, and that...

SUBJECT: Yes, but that's going down there. [indicates lower on an hierarchical tree diagram]

INTERVIEWER: Well, at this point, we haven't said you're only allowed to do two moves.

SUBJECT: Could the diamond be below the circle? Could the diamond not be below the circle, did you say?

INTERVIEWER: Could the circle be below the diamond?

SUBJECT: Oh right, sorry.

INTERVIEWER: Not in this bit.

SUBJECT: But if you went further. Do you mean... Yes, I'm sure it could.

INTERVIEWER: Right, so what we've got is two different claims here. For one of these, it's impossible, and the circle must be above the diamond. But if we change what we're looking at slightly by looking even further forward...

SUBJECT: Yes, go ahead...

INTERVIEWER: ...you've got that that condition is no longer true. Right? So whether or not it's true depends on how you're looking into the future, whether you're just looking at the next step or if you're looking further forward.

SUBJECT: Yeah.

INTERVIEWER: Ok. So let's...

SUBJECT: I sort of got that from there. Very brief... very the gist of it, sort of. Go on.

INTERVIEWER: Well, you go on. Tell me what's happening.

SUBJECT: Well, I just think you could make it, you could just sort of go on, couldn't you.

INTERVIEWER: Yeah, and each of those would have... well, let's see...

SUBJECT: Another box.

INTERVIEWER: Well, each of those would have two possible things you could do, wouldn't they, because there's only two spaces you could move from. So each of those would...

SUBJECT: So you could just go on generating these.

INTERVIEWER: You can if you want. Ok, so, we've got this system here, "[Subject's name's system of Modal logic." And you've managed to make claims about this system here. What you've managed to make are Modal statements about possibility and necessity.

SUBJECT: Yes.

INTERVIEWER: So you understand that?

SUBJECT: Yes.

INTERVIEWER: And you see that if we extend these further, then depending on how you allow that to be extended...

SUBJECT: It could go back to that.

INTERVIEWER: It could go back to that, yes. But depending on how you allow these to extend, you're going to get even more claims. You're going to get differences between the claims. Something which is true in here won't be true in general. Well, let's look at how it can change. What sort of questions might you ask. Well, we've looked at this one. What does that really ask? Does this have to hold... when?

SUBJECT: Actually I don't know if you can actually say that the circle will always be over it, I don't think it's in any of that.

INTERVIEWER: Because if we extend it you can rearrange that?

SUBJECT: Yes, yes.

INTERVIEWER: Ok, but you can say it at this level.

SUBJECT: Yes.

INTERVIEWER: So what's the condition that means you can say it at this level?

SUBJECT: Oh right — you want me to say what conditions here are true?

INTERVIEWER: Let's take the statement, let's write this one down: "The circle must be above the diamond."

SUBJECT: That's the statement.

INTERVIEWER: That's your statement.

SUBJECT: And I have to say whether...

INTERVIEWER: Right, now, hang on. What you're trying to do is, you're trying to prove whether this is a valid or an invalid statement, whether it's true or not. Ok? Would it be a valid conclusion to say that that's true, given this?

SUBJECT: Ok.

INTERVIEWER: And you've got this as you're accessibility relation...

SUBJECT: Right, am I just looking at this time, or...

INTERVIEWER: Now, that's the question! What conditions must hold, given this, given that, to show whether that's valid or not.

SUBJECT: I don't know what...

INTERVIEWER: Well, you've just said it, really. "Are we looking at this step, or are we..."

SUBJECT: Oh, right. Yes.

INTERVIEWER: And you can see there's a difference coming out there, yes?

SUBJECT: It holds here, but...

INTERVIEWER: Ok. So write down that it's valid for one step, it's not valid if we look more than one step, and it's not valid if we just ask something more general, like, "can it ever happen that..."

SUBJECT: Something more general...

INTERVIEWER: That'll do. So, we've got a difference between this one and that one. There's the specific case, and we've got the more general cases. This specific one is T.

SUBJECT: Aahh!!

INTERVIEWER: Right. These more general ones, we'll split them up in a second, are the S systems.

SUBJECT: That's what that is.

INTERVIEWER: Right. And you've spotted this on your own by saying, "well, it's true here, but not if we look into the future."

SUBJECT: Yes.

INTERVIEWER: Ok. Well, let's try and think about what the next step should be. In your system it's going to be very hard to differentiate between these because you can extend it indefinitely and you can shuffle round corners. From any one of these you can get to any of the others if you look far enough forwards. Let's say we limit this. Let's say we limit this to at most three moves.

SUBJECT: The max... what, for each object?

INTERVIEWER: No, in total. It doesn't matter which one moves, but we're only going to let something move three times.

SUBJECT: Right. Ok. So the diamond can only move three times.

INTERVIEWER: Well, actually, let's say we will only allow it to move twice, and then we can draw this one out, can't we.

SUBJECT: Ok.

INTERVIEWER: Because you're only going to get one level further down than that aren't you.

SUBJECT: So the triangle can only move twice. I see! So we've got a finite... space. So we can... Ah, for each of these comes another layer.

INTERVIEWER: Yes, so you start off with your the top one.

SUBJECT: So we've got three, but that one you know.

INTERVIEWER: But draw it in at the top anyway, so that we can keep track of everything that's going on.

SUBJECT: Then this will come out.

INTERVIEWER: They will each go out to...

SUBJECT: Ooh... circle... diamond... do you want me to do this? Do philosophers use logic?

INTERVIEWER: Yes.

SUBJECT: Thought they might.
INTERVIEWER: [laughs] For any reason?
SUBJECT: I did philosophy of mind [indistinct]. Because I'm sure they talk about possible worlds, but it might be in a slightly different way.
INTERVIEWER: It tends to be linguistic philosophy, but yes. They do.
SUBJECT: [indistinct] This is talking about whether we have minds or not. It's so relevant. Right. And this one... oh, we know they can only move twice. So in this one we could get... that one moving back there, so that'd look like that. Or we could get... I'm assuming there will be three in each time. Or that triangle could move there, yes? Any more? No. I thought that could go to three. And this one... I don't know why I chose this noughts and crosses... the circle could move there, or the circle could move back up again. And this one... reminds me of search this.

INTERVIEWER: Search?
SUBJECT: Yeah. Have you come across search in AI? It really reminds me of that. Can I tell you?

INTERVIEWER: Yes.
SUBJECT: You have a problem, and you can define it, let's say simple like erm... I don't know, think of a good example, erm... games; give me a game...

INTERVIEWER: Chess.
SUBJECT: No, that's not a good one.
INTERVIEWER: Noughts and crosses.
SUBJECT: Oh no, route finder's quite a good way. So you have... you'd have three routes, but basically if you want to get from A...

INTERVIEWER: Oh right, like maps and networks and stuff.
SUBJECT: Yeah... you can do it in a search way, so you can go A... you could do... oh, I'm not explaining very well, I'm getting confused already... [discusses route solving problems.]

INTERVIEWER: Is that something you would have thought of before looking at that material?
SUBJECT: Oh yes.
INTERVIEWER: That reminded you of the search.
SUBJECT: Yes, this did.
INTERVIEWER: But was it useful to be reminded of the search?
SUBJECT: No, probably not.
INTERVIEWER: Why?
SUBJECT: Because it's from a different... it has different procedures and so on.
INTERVIEWER: It's not at all transferable as something that could be useful background?
SUBJECT: No, I don't think it is. No, what it basically is is that the tree would be used to represent a certain problem space, and there's certain procedures or strategies which work through that problem space... [explains]
[back to problem.]
And this one could move... diamond could move down again. Or the diamond moves sideways. I think that's it.

INTERVIEWER: That is it. So we know from this one that there's a difference between the claim, can the circle be below the diamond, and can it ever be below the diamond. Right, now what we've got here, is that we've got a lot of duplicated ones. Let's say on this, this... if we look forward and say, can we get this at a different stage... Let's try and find one that's useful as an example. Let's say can the triangle be between the circle and the diamond.

SUBJECT: [writes it down]
INTERVIEWER: I can here, but it can't here.
SUBJECT: Can the triangle be between the circle and the diamond...
INTERVIEWER: Well, I suppose it depends on what you mean by between. But if we're talking about it being in a directly straight line...

SUBJECT: There.
INTERVIEWER: Yes. So there's that one. If we allow it to move twice, it can't.
SUBJECT: No.
INTERVIEWER: So... we could ask, "can the triangle be between the circle and the diamond?", and we'd say yes, "can it be the case that it can be the case that the triangle is between the circle and the diamond."

SUBJECT: Yes.
INTERVIEWER: Where.
SUBJECT: There.
INTERVIEWER: Can it be the case... that it can be the case...
SUBJECT: Yes.
INTERVIEWER: Ok. What does it mean by "can", then?
SUBJECT: Might possibly be...
[tape break]
INTERVIEWER: Can it be the case... that something can happen... So we're adding another layer onto that.
SUBJECT: Oh, right.
INTERVIEWER: That's what I'm trying to distinguish here. And in that case, no, it can't.
SUBJECT: Cause you've just said can it be... and I said yes, but I was thinking of it in another way...
INTERVIEWER: Yes, and that's exactly the point. How you interpret "can"; what's going to hold.
SUBJECT: Ah...
INTERVIEWER: So the conditions you're allowing to happen affect what's going on. Strictly, on the accessibility relation you've done, what I'm saying is, is it possible that "(possible (possible p)) is not true, but that possible p is true". Now if you change the relationship so that it doesn't have to move...
SUBJECT: So that describes the "can it be (can it be)"... Ahhh... I'm getting this. Yes.
INTERVIEWER: And if you change your accessibility relationship so that that doesn't have to move, it'll change, because you can stay there. So that will be true and that will be true.
SUBJECT: Yes. So if I just change my conditions here...
INTERVIEWER: Then the structures are going to alter. So what we come out with is... let's try and think of another one.
We've got some distinction between one operator and two operators. We also have... well, can it be the case that these things must not be in a straight line.
SUBJECT: I can never understand negatives like that. Can it be the case that...
INTERVIEWER: That these things aren't in a straight line. Ok, let's put it another way. Can it be the case that nothing is in the middle square. Well, not on any of these. So on this one... let's write this down. [writes it] Well, what's the next step? Can it be the case that it can be the case that nothing's in the middle square?
SUBJECT: Yes... loads! The first...
INTERVIEWER: So possible possible p is true.
SUBJECT: So it was false here because we've got them all in the middle square.
INTERVIEWER: Now, can it be the case that it must be the case that nothing's in the middle square?
SUBJECT: Can it be the case... it must be the case... no.
INTERVIEWER: Why?
SUBJECT: Because by saying must, you're saying in all situations, and none of those situations...
INTERVIEWER: Ok, right, but what about these levels?
SUBJECT: What about?
INTERVIEWER: The levels.
SUBJECT: Well, you only said one "can".
INTERVIEWER: Yes, but can it be the case that it must be the case... Ok, can it be the case that something's in the middle... yes. Can it be the case that nothing's in the middle... no. Must something be in the middle? Let's start by writing these down. [writes] I'll call this possibly S, and this possibly N, for something and nothing.
SUBJECT: Ok. Say that again.
INTERVIEWER: Can it be the case that something can be in the middle.
SUBJECT: Yes.
INTERVIEWER: Show me.
SUBJECT: Here.
INTERVIEWER: Ok — what's the difference between that and "can it be the case that something is in the middle"?

SUBJECT: What's the difference between that and... you've got another can. So you're looking down another level.

INTERVIEWER: Yep, because here were taking can in fact to mean "After the next move". Each one is looking at the next step along. So can it be the case that something can be in the middle.

SUBJECT: Yes.

INTERVIEWER: Show me.

SUBJECT: Can it... Ah! No no no! Not here!

INTERVIEWER: So that one's false. So can it be the case that nothing's in the middle? Can it be the case that nothing can be in the middle?

SUBJECT: True.

INTERVIEWER: Must something...

SUBJECT: Hang on... this is the tricky one. True.

INTERVIEWER: Why?

SUBJECT: Because they're all in the middle. Must. It's a must.

INTERVIEWER: Must nothing be in the middle?

SUBJECT: False. Now you're going to take me to the next level.

INTERVIEWER: Yup. How shall we do it? With a can or a must?

SUBJECT: Do a must. Because I think I've got the hang of "can".

INTERVIEWER: Must it be the case that something must be in the middle.

SUBJECT: False. It is false, yes.

INTERVIEWER: Because?

SUBJECT: Because there's nothing in the middle of those. It's false. If there was one...

INTERVIEWER: Ok. Must it be the case that nothing is in the middle.

SUBJECT: Yes, according to this diagram.

INTERVIEWER: Can it be the case, that the triangle must be on the right?

SUBJECT: You've got a can and a must.

INTERVIEWER: I have indeed.

SUBJECT: False.

INTERVIEWER: Why?

SUBJECT: Because you're saying the triangle must be on the right, and it's not always.

INTERVIEWER: Ah, but if it was always, it'd be must. It'd be "must it be the case that it must be on the right".

SUBJECT: [laughs] No — because it's the must by the triangle.

INTERVIEWER: Ah, but if you're here, then it must be on the right. If you're at that point. So can it be the case that it must be on the right? Ok — must it be the case that it must be on the right?

SUBJECT: False.

INTERVIEWER: Ok. Because...

SUBJECT: Because there are these situations where it's not.

INTERVIEWER: Ok. Can it be the case that the triangle must on the right. Can it be the case after one step that the triangle must be on the right after the next step.

SUBJECT: That doesn't make sense.

INTERVIEWER: It's putting in your accessibility relationship. Can it be the case after one movement... So which ones can it be after one movement?

SUBJECT: Those two.

INTERVIEWER: So two can. All of these can happen. But in addition, with these two, if this happens, then this must also happen afterwards.

SUBJECT: Oh yes.

INTERVIEWER: So can it be the case that the triangle must be on the right?

SUBJECT: True.

INTERVIEWER: But it doesn't have to be the case that the triangle must be on the right.

SUBJECT: No, because you've got this one.

INTERVIEWER: Now, what if we said, ok, that's saying, this is breaking this down step by step, and you've got this idea of Modal operators, haven't you.

SUBJECT: Yes. When you say Modal operators...

INTERVIEWER: Must and Can.

SUBJECT: Operating on different things. Yes.

INTERVIEWER: Based on this, let's tweak what we can look at. Can the triangle ever be on the right?

SUBJECT: Yes.

INTERVIEWER: Must the triangle always be on the right?
 SUBJECT: No. You can say quite a lot, can't you.
 INTERVIEWER: Yes, even with something this simple.
 SUBJECT: You can. Even I can get it! Right.
 INTERVIEWER: Ok, if we allow you to look two steps into the future, as well as just one step at a time, can it be the case within two moves that all the symbols are in a straight line?
 SUBJECT: No.
 INTERVIEWER: Why?
 SUBJECT: Because they're not there. Oh — within two moves? Yes, there. Yes. You've got to interpret this carefully.
 INTERVIEWER: That's the point. It's interpretation. Ok. Well, those are the different things... Can you see what's happening there?
 SUBJECT: Yes. Quite a useful tool, I'm sure.
 INTERVIEWER: So what we've got, is that we've got system T.
 SUBJECT: Is that system T?
 INTERVIEWER: Well, what we've looked at it all three.
 SUBJECT: Oh!
 INTERVIEWER: Next step. Right, what will happen next? That's why you get those symbols stacked. S4...
 SUBJECT: Why is it called S4? And why do they call it system T?
 INTERVIEWER: T was the most rudimentary one they could come up with.
 SUBJECT: Why didn't they just call it...
 INTERVIEWER: Fred?
 SUBJECT: Yeah, T and S4 — I mean, they're just being clever.
 INTERVIEWER: [explains about numbered systems; says about properties]
 SUBJECT: Oh right. Oh!
 INTERVIEWER: What will happen to the structure of our diagram... let's see, you've got the grid here, and the three down here, and six down here... Right. Now, in system T, you've got this.
 SUBJECT: One step at a time. And you can go on doing that?
 INTERVIEWER: You can do it as often as you want.
 SUBJECT: I like T.
 INTERVIEWER: What about if we say, within a certain amount of time, or a certain amount of moves. What's going to happen to these diagrams? We can jump forward steps. We don't have to look at them one at a time. So we'll have something which looks like this.
 SUBJECT: And you can get programs to generate this? I think you could.
 INTERVIEWER: So you can get this, like the one we had before, or you can go straight to it.
 SUBJECT: But you're not saying by saying that that you can get from there to there.
 INTERVIEWER: Within two moves you can.
 SUBJECT: Yeah, within two moves.
 INTERVIEWER: But not in one. Well, this is fair enough, but if you actually look at these, some of these are the same as the one you started off with. So what you do is, these ones are all different, so you leave those in. And then...
 SUBJECT: But then little arrows, you use an arrow, you use an arrow to go back! Yes, I remember that bit.
 INTERVIEWER: Right, that one's just had to stay where it is, we haven't had to worry about that.
 SUBJECT: Oh right. I knew there was something significant about the arrow swooping back.
 INTERVIEWER: Well, the reason it's significant...
 SUBJECT: Oh, it's this one!
 INTERVIEWER: ...is that you get these loops.
 SUBJECT: That could go on forever.
 INTERVIEWER: It could indeed. So this one could go back up to here, or it could go on to another one which is down here. In fact, you'll find all three of them can. They can all go back, or go on to another state.
 SUBJECT: Yes, because some of those were the same.
 INTERVIEWER: Yes, in fact there's one in each condition that's going to be the same, because you can move back where you've just come from. So what will actually

happen is, if you get these loops, you'll find you can cancel down these musts and cans, because if it's, "within a certain amount of time", you don't have to worry about what each of the next steps is going to be. You're just looking at whether you can get there or not.

SUBJECT: Yes.

INTERVIEWER: Ok, so that's that one, and you end up with your seven boxes instead of ten.

SUBJECT: Oh I see — you mean you don't really need to know that it'll go back to being what it started with.

INTERVIEWER: You don't need to know where it's been, you just need to know where it's going to get to.

SUBJECT: Yes, exactly. So this is cutting down. If that and that are the same, you don't really need to know that, do you? What I mean is, all it's told you is that it's...

INTERVIEWER: Hold on; in system T, if that and that are the same, you don't need to worry about whether or not they're the same, because you're also thinking about where you've been to get there.

SUBJECT: Yes; right.

INTERVIEWER: In system S4, you're not worried about where you've been in the same way; you're worried about where you've arrived. So if it is the same, you can simplify it by pointing it back upwards.

SUBJECT: Because you've been there before.

INTERVIEWER: Yes. Now, say we allowed three movements, we could actually end up further down here, one two three, or up here, one two three. So if we have three movements, it might be the case that we could cancel down just to whatever happened for this one. So say this diagram here also cropped up down here; we could go one two three [indicates a three-operator movement].

SUBJECT: Which diagram?

INTERVIEWER: This one. Let's call it "hello". Let's say "hello" crops up down here as well. So we can go from here to here, once to this initial world again, and on down to "hello". Right?

SUBJECT: Yeah.

INTERVIEWER: But say we want to keep the diagram simple; instead of going on to this new one and on to down there, we could go once there, once back, and once to "hello" over there. Which means, in effect, instead of having three movements, we only need one. So if we had three operators in a row, we would probably no longer need those three, we could cut two of them out.

SUBJECT: Yes, yes! Those operators you use to... yeah, yeah, yeah.

INTERVIEWER: What this system lets you do is cancel those down. And the cancellation comes because you can go back up, which is equivalent to just doing a shorter journey. Right; you got that?

Now, the other thing you can do is say, can it ever happen (in this time)? And what you've got is these seven in S5, but instead of having loops forwards and up, you just link all of them to every other one.

SUBJECT: Ah, like this one! [points to instructional material]

INTERVIEWER: Yup.

SUBJECT: But it got a bit mad, didn't it?

INTERVIEWER: It did, yes, which is why instead of drawing in every line we just drew the ones around the edges, and said you can travel along as many of them as you want. Alright?

SUBJECT: Yes.

INTERVIEWER: Now what we're saying here, is, "can you ever get to...", and we know that within two moves you can get to all of these. We don't care how you get there, it's just whether you can or not.

SUBJECT: Ah... because it doesn't matter how you get there. Not in this... It might matter in a...

INTERVIEWER: It depends what you're looking at to try and achieve. But here you're just looking at the end result. Can we ever get to that state?

SUBJECT: Ahh!

INTERVIEWER: Right, this means we can get to any of them in one journey.

SUBJECT: Yes, because we've just linked everything up!

INTERVIEWER: So instead of having three of these things, we don't even have to worry if it repeats a pattern, we can just cross out all but the last one, because we've only got one journey. So... differences. This one builds them up one at a time. This one you can cancel out if you do the same thing again and again, so any repeated patterns go. And this one, S5, you can cancel out all but the last one.

SUBJECT: So why didn't they start with that one and that one in the first place?

INTERVIEWER: Well, because this one actually takes on a whole load more assumptions, like, "we don't care how you get there." This one is the most specific in terms of all the information that it keeps.

SUBJECT: Ok.

INTERVIEWER: So, in general, this one will tell you the most, and then you can whittle it down to one of the other ones. You can't work it back up to get back to that one, though. Do you see what I mean?

SUBJECT: Yes.

INTERVIEWER: If you've given that, you're not told anything, are you? But if you have the other one, you can then cut it down to that.

SUBJECT: Yes, because you can cancel things off of it.

INTERVIEWER: So, T, S4, S5, and the differences between them are that...

SUBJECT: T, you can say what's going to happen at each step, S4... there's S4 and S5, aren't there?

INTERVIEWER: Yes.

SUBJECT: S4 you can... well, it's to do with what you can say will happen within two steps?

INTERVIEWER: Well, here you can...

SUBJECT: Within a certain time.

INTERVIEWER: Yes. And if you get repeats, then...

SUBJECT: You just go back. You cancel stuff out.

INTERVIEWER: And S5 you can...

SUBJECT: You can just go to wherever you want.

INTERVIEWER: And what does that mean with the operators?

SUBJECT: You don't say them... you've done it here...

INTERVIEWER: Except for... one journey, so you leave in one.

SUBJECT: Ok. So you would have one for one step down.

INTERVIEWER: And in here, the reason you can cancel them down is because taking three steps is the same as taking one. Like, two steps forwards, one step back, is the same as one step forwards.

SUBJECT: Ok, but there's different reasons behind why you only use one operation in those two.

INTERVIEWER: Ok, well, let's give you a really messy example in terms of operators, then. Let's give you this one... Right, ok?

SUBJECT: What's the square?

INTERVIEWER: Square is necessity, must be the case. And the diamond is possible, it can be the case.

SUBJECT: Oh, right. So...

INTERVIEWER: So in T, what would that become?

SUBJECT: In T... what do you mean, what would it become?

INTERVIEWER: Can you simplify that in T?

SUBJECT: Erm... no.

INTERVIEWER: Can you simplify that in S4?

SUBJECT: Erm... yeah, you can cut out three. You said you can cut out three to make one. I don't know what you do when there's more than three.

INTERVIEWER: Ok, well, why can you cut out three to make one?

SUBJECT: Because you said you could get somewhere in one step or three steps. Get to the same place in one step or three steps. The same state as such.

INTERVIEWER: Because you're actually repeating it, yes? What you're doing is you're going one two three, effectively, you're repeating that loop.

SUBJECT: Yes, ok.

INTERVIEWER: So any repeats you can cut out.

SUBJECT: Oh, so we can cut these out.

INTERVIEWER: Right; is that a repeat of this?

SUBJECT: They're musts.

INTERVIEWER: Yes, but these are musts together. This is a must... this is a must can... if you have must must, it's repetition, if you have can can, that's repetition. If you have must can must can, that's also a repetition. It's like going, one, two, and back one two.

SUBJECT: But this is talking about must cans.

INTERVIEWER: Yes, but each of those is the equivalent of a step. So, in S4, what could you cancel?

SUBJECT: Oh, we're on S4 now?

INTERVIEWER: In T you can't cancel.

SUBJECT: Well, in S4 you've got two musts...

INTERVIEWER: Right, so cancel one of those, and get...

SUBJECT: Can't you cancel both?

INTERVIEWER: Cancel them down, not take them out.

SUBJECT: But you've got one here! No, sorry. No, no, I'm being silly. I'm trying to see these as steps in the process. So you've got two... I'm not clear why I cancel that one.

INTERVIEWER: Let's say here... you've got three here, and we'll call that... let's not bother with it... ok, let's call that must might must. You go forward and you get another might and another must and another might and another must, and then you go onto something else. Right. What this would be is you get one two one, another diamond and another square, and another one depending on how many times you go around the loop, but each of these diamond squares is the same pattern. Yes? No?

SUBJECT: No.

INTERVIEWER: Right, back to this.

SUBJECT: I got that.

INTERVIEWER: You got this? Let's look at your one so that we can see what all the symbols mean. Can it be the case that they're in a line.

SUBJECT: Oh, I'm stuck on the crossing out.

INTERVIEWER: Yes. I'm going to get to the crossing out. Can it be the case? Yes. Can it be the case if we're allowing it to go backwards again. Can it be the case that it can be the case that they're in a line. No. Because you either go forwards, in which case none of them are true, or you go back, in which case it's not true. Now, can it be the case that it can be the case that it can be the case that they're in a straight line? Forward one, and then either back one or forward one; yes, you can do it that way. Or forward one... you can do it that way. And the reason you can do it, is because it's the same as this one step forward. So instead of three of these, you only need to write one. It's the same thing.

SUBJECT: Oh. I think I get that.

INTERVIEWER: So what you're doing is, instead of looping around in lots of circles, you're only going around each loop once. It doesn't matter how many times you go around it. If you're in T, you have to say forward once, back once, forward another time. If you're in S4, you don't worry about that.

SUBJECT: Ok, I think I get that. I think my mind's going.

INTERVIEWER: Well, just have a quick go at that.

SUBJECT: Well, you said, basically, two musts... see I'm not sure where I'm cancelling it out, really. I mean, I understood why you did it there, so I assume it's the same reason.

INTERVIEWER: It's exactly the same reason.

SUBJECT: Ok, well I'll just assume that, and not...

INTERVIEWER: Ok, so you've crossed out one of these musts, because you've cancelled that down because you had two of them.

SUBJECT: Ok, right, yeah.

INTERVIEWER: Right, so now you've got must might must might.

SUBJECT: That's all repetition.

INTERVIEWER: Right, so you can cancel what?

SUBJECT: A must and a might.

INTERVIEWER: Yes.

SUBJECT: See!

INTERVIEWER: So you're left with must might p.

SUBJECT: Right...

INTERVIEWER: And that's in S4. Now what about if you're in S5?
SUBJECT: ooh...
INTERVIEWER: Ok, go back and look at the diagram. It's this one.
SUBJECT: No, hang on — I'm just looking at that. You said you could get anywhere. One. You can get anywhere in one. So you just cancel them all out.
INTERVIEWER: Except...?
SUBJECT: Except...
INTERVIEWER: You still need to take one step.
SUBJECT: And what step would it be? Would it be the must or the can?
INTERVIEWER: It would be the last one.
SUBJECT: The can. It's always the one that it's next to.
INTERVIEWER: Right, so there you've got it. You've got these things, and under different conditions, you can say, this is a valid conclusion if we're in S5...
SUBJECT: [laughs] Cor!
INTERVIEWER: This one's a valid conclusion if we're in S4, or if we're in S5. You don't have to cancel down as far as that.
SUBJECT: Yes.
INTERVIEWER: And this one, just leaving it on its own, is valid in T, S4 and S5, because we don't have to do any cancelling, we just leave it as it is.
SUBJECT: Yes.
INTERVIEWER: Ok. Let's have another go at the post test. We'll do this in another colour. Just have a look at this, see what you can do.
SUBJECT: Oh, goodness. In red?
INTERVIEWER: Just see what you can do. Red's fine. So (a). Ok, if you're given can can p.
SUBJECT: But you don't know which system...
INTERVIEWER: What it's saying is, under which systems can you say "can p".
SUBJECT: Oh! Under which systems can you say this? You can say it in T...
INTERVIEWER: Can you? Why? ...Ok, ignore this middle sign here [implication]. You're given the start. You're told this bit, that "can can p" is true. What you want to know is, when can you just say that. When is this equivalent to just saying that.
SUBJECT: I don't understand what this is asking.
INTERVIEWER: Right, what this is asking is if you're given something like this, this string here at the start, what can you say about it at the end, depending on which of these you're in? What it's saying is, if you're given...
SUBJECT: What can you do with them?
INTERVIEWER: Yes. What you can say is, you can get that answer out if you're in each of the following systems...
SUBJECT: Ah! I see, I see! Sorry! What you can... sorry. Well, you can't do anything with T, and...
INTERVIEWER: Can you do it in S4?
SUBJECT: No.
INTERVIEWER: Because?
SUBJECT: You need three to make on in S4.
INTERVIEWER: You just have to have a repeated pattern.
SUBJECT: I thought that was... Oh, I'm sorry.
INTERVIEWER: No, it's alright. It's my bad explanation. Because what I should say, you can say "is it the case?" as well as can and must. Is it the case here, in the starting world, that the triangle's on the right hand side.
SUBJECT: Oh, I'm getting lost again.
INTERVIEWER: What it's saying is that, in S4, is if you've looped back to the start, you've got a loop, you don't need to have taken any steps. You need one operator per step you take.
SUBJECT: So that's a repetition so you could cancel that out, and that could be S4.
INTERVIEWER: Can you do it in S5?
SUBJECT: Yes. Because you can get anywhere in S5.
INTERVIEWER: Now, don't do (b), because it's a bit of a trick question.
SUBJECT: Now this one is a bit similar to the one you've shown me. Ok, I reckon in...
INTERVIEWER: Can you do it in T?
SUBJECT: No. No you can't.
INTERVIEWER: Can you do it in S4?

SUBJECT: Sorry, I'm grasping at straws here. Yes! You can do that in S4!

INTERVIEWER: Can you do it in S5?

SUBJECT: Erm... I don't know.

INTERVIEWER: If you can do it in S4, you can do it in S5, so yes you can. What about this one?

SUBJECT: Ok...

INTERVIEWER: Now, you're allowed to just take that symbol inside the brackets.

SUBJECT: Yeah, just... Isn't there a law about this?

INTERVIEWER: It's an algebraic sort of thing. You just take that inside. Because really that's part of first order logic. Now, will that be valid in T?

SUBJECT: I think so, yes.

INTERVIEWER: Ok, so we'll put T. Why?

SUBJECT: Because if... p and q can be... it's hard to say it, but you know what I mean. That could mean "Can p and can q".

INTERVIEWER: You don't have any problem with cancelling down.

SUBJECT: Well, there's no repetition here.

INTERVIEWER: So you don't need to cancel, so that will be true.

SUBJECT: I don't think I get that last one.

INTERVIEWER: Ok, given the cancellation rules, ok...

SUBJECT: Yeah.

INTERVIEWER: Do you need to cancel for this one?

SUBJECT: You can't. There's nothing to cancel.

INTERVIEWER: So is it valid in T?

SUBJECT: Yes.

INTERVIEWER: Is it going to be in S4?

SUBJECT: Yes. Because it's going to be true in all of them.

INTERVIEWER: Right! No problems. Now, question 4.

SUBJECT: Oh! I want to go!

INTERVIEWER: This'll take five minutes. You either get it or you don't.

SUBJECT: [reads question]

INTERVIEWER: Ok, so draw that. Draw a world.

SUBJECT: It's only related to itself.

INTERVIEWER: Right. Is that it?

SUBJECT: Yes.

INTERVIEWER: That's fine.

SUBJECT: We'll have a couple of them. Is that enough?

INTERVIEWER: Well, we only need to look at one.

SUBJECT: Ok... [indistinct] [reads question]

INTERVIEWER: Show that, if you have "it must be p", you can say "p".

SUBJECT: Either way.

INTERVIEWER: Either way round. So, start with "it must be the case that p".

SUBJECT: I don't get this.

INTERVIEWER: Start with "it must be the case that p".

SUBJECT: Oh no — I've just got a case of deja vu here...

INTERVIEWER: It's not "Can be the case."

SUBJECT: Oh sorry... I'm getting... I know these seem basic, but it's amazing...

INTERVIEWER: Ok, if it must be the case that p, what can you say is true in this world?

SUBJECT: In my notes... I don't know.

INTERVIEWER: Ok, what does it mean if it must be the case?

SUBJECT: Erm... I don't know! It means all sorts of things!

INTERVIEWER: Like?

SUBJECT: I don't know! Is this to do with statements? I don't know what you mean.

INTERVIEWER: Let's look at your example. Draw one of these. Just one of them. Ok, this is your starting world. Now it's only related to itself, which means it can't move anywhere.

SUBJECT: Ok.

INTERVIEWER: If we say it must be the case that the triangle is right of the circle, what can we then say? That it...

SUBJECT: That it's true.

INTERVIEWER: Yes! So you just say "the triangle is right of the circle." If it must be the case, then it actually is the case.

SUBJECT: Yeah. I don't know what I'm showing. I'm showing this? How do you show that?

INTERVIEWER: What you do is, you start with one side, and show the other one, and then you start with the other one, and show the first side. You assume each side is true in turn, and show the other one. So you assume that this is true, that the triangle is right of the other two. If we're assuming that's got to be true, what can we say?

SUBJECT: I don't understand.

INTERVIEWER: Right, the statement here, say, that is right of... that's p.

SUBJECT: Oh! Right.

INTERVIEWER: So if we say it must be the case that the triangle is right of the circle, what can we say? We're given this, that the triangle is right of the circle.

SUBJECT: I'm really lost. I really don't understand this. I don't understand what I'm doing here.

INTERVIEWER: The proof? Right. What you do is first of all you assume that it must be the case that this is true...

SUBJECT: But how do you show it?

INTERVIEWER: Normally you build up a system of statements, each of which is derived from the previous one, and at the end of them all you get to the other.

SUBJECT: So what do you start with here?

INTERVIEWER: You either start with the left hand side or the right hand side and then you show the other one.

SUBJECT: Ok, I've got to show that p is...

INTERVIEWER: You've got to show that if p is true, it must be the case that p...

SUBJECT: If and only if...

INTERVIEWER: Which splits into, "if that is true, then this," and "if this is true, then that."

SUBJECT: I can't do it.

INTERVIEWER: Ok; don't worry about it. Thank you! Sorry it went on for ages.

SUBJECT: No, it's alright. Sorry I'm confused!

INTERVIEWER: That's alright. Do you think that you learnt anything from that?

SUBJECT: Oh yes! I did! I did!

INTERVIEWER: Did you think that that was a useful way to go through it?

SUBJECT: I did. I got a bit... and it might be because I'm saturated... because you do... I certainly understood all that.

INTERVIEWER: All the different systems and how they work.

SUBJECT: Yeah, basically.

INTERVIEWER: I mean, you've got as far pretty much as the cancellation.

SUBJECT: Yeah. That's probably not very far, is it?

INTERVIEWER: It is actually. It's as far as most people get at all. The question 4 is a transfer of knowledge one.

SUBJECT: Is that why you were pushing me?

INTERVIEWER: Well, I was wanting to see if after talking through with you...

SUBJECT: I think I'm a bit scared of this sort of thing.

INTERVIEWER: Well, I know, I could tell that. [explains q4]

INTERVIEWER: Oh, just quickly, I'll show you the other two, and you can say which one you'd have preferred.

SUBJECT: How do you decide who gets what?

INTERVIEWER: One two three one two three... And in this one, you get...

SUBJECT: These are just different ways of explaining it.

INTERVIEWER: Yes. Same information in each one.

SUBJECT: I think I'd have liked my one actually.

INTERVIEWER: Right. Why?

SUBJECT: It's got pictures in it! You know, it's got those things.

INTERVIEWER: Those things? The grid worlds?

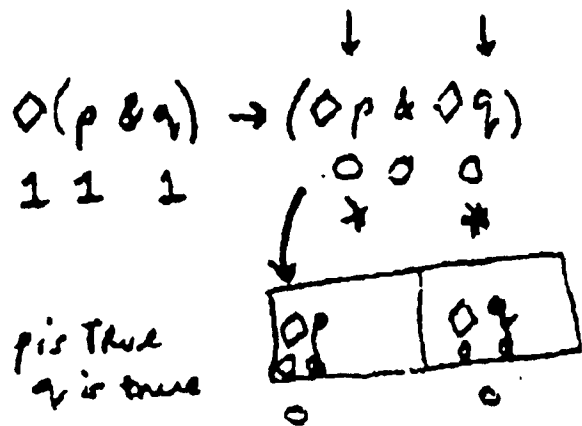
SUBJECT: Not that they threw me a bit, but they obviously helped a little bit.

Appendix 5: Workscratchings, first study

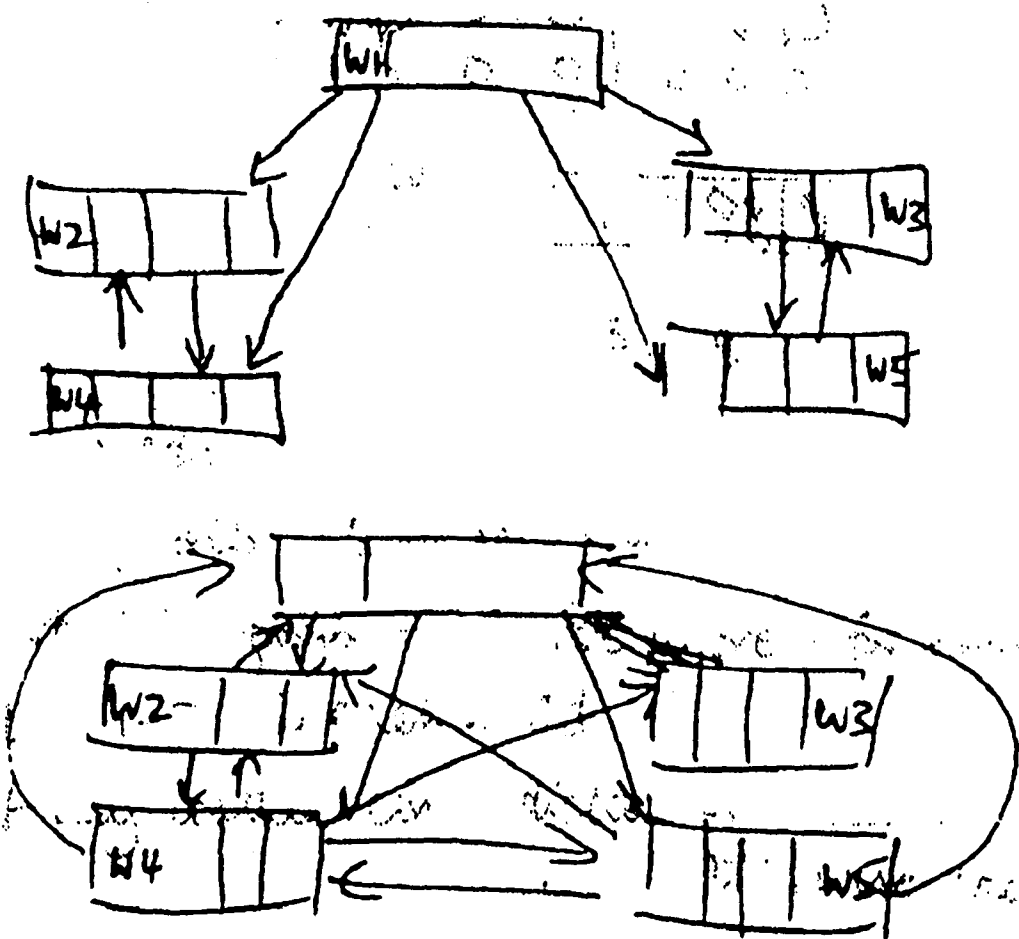
This appendix collates students' diagrammatic workings from the intervention exercises, pre-tests and post-tests of the study described in chapter three.

Diagrammatic condition

Subject 17



Subject 23



$$p \leftrightarrow \Box p$$

1 1 1

$$p \leftrightarrow \Box p$$

0 0 0

$$p \leftrightarrow \Box p$$

1 0 0

$$W1 \quad \Box (p \vee \Diamond q) \rightarrow (\Box p \vee \Diamond q)$$

1 1 1 0 0 0 1 0 0 0

$$W2$$

\Box	$(p \vee \Diamond q)$	\rightarrow	$(\Box p \vee \Diamond q)$
0	0 1 0 0	1	0 0 1 0 0

$$W3 \quad q$$

1

Subject 26

$$W1 \quad \Box (p \rightarrow q) \rightarrow (\Box p \rightarrow \Box q)$$

$$W1 \quad \Box (p \rightarrow q) \rightarrow (\Box p \rightarrow \Box q)$$

$$W2 \quad \Box p \rightarrow \Box q$$

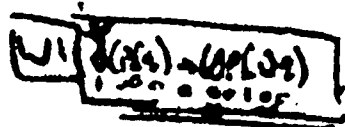
$$W1 \quad \Box (p \rightarrow q) \rightarrow (\Box p \rightarrow \Box q)$$

$$W2 \quad p \rightarrow q$$

$$c) \Box \Diamond \Box \Box \Diamond \Box p \rightarrow \Box \Diamond \Box p$$

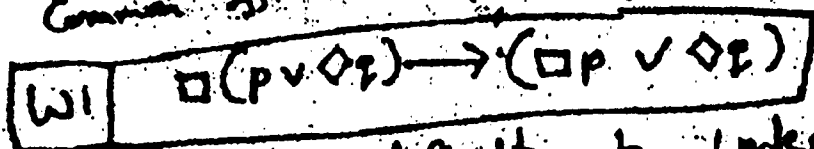


$$d) \Diamond(p \& q) \rightarrow (\Diamond p \& \Diamond q)$$



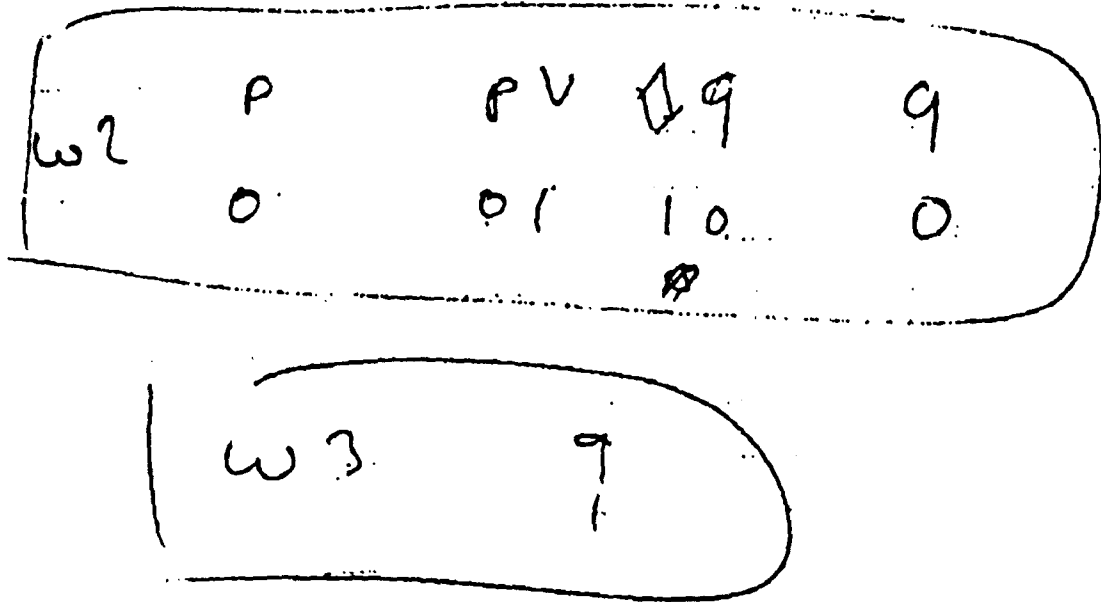
Subject 28

Common Systems are T, S4 & S5



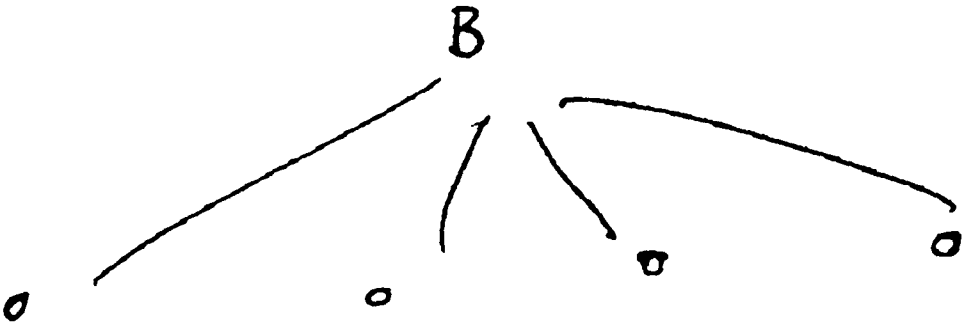
is difficult to understand

Subject 31

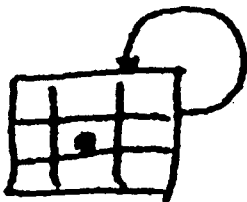


Learning Environment condition

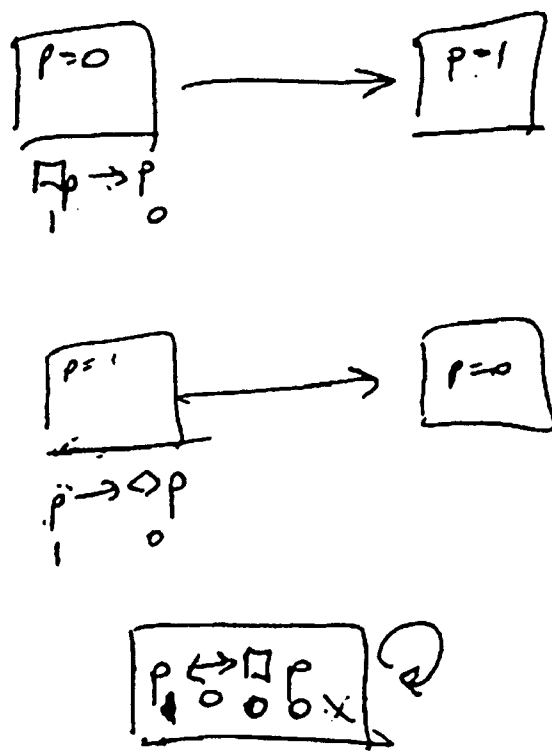
Subject 38



Subject 40



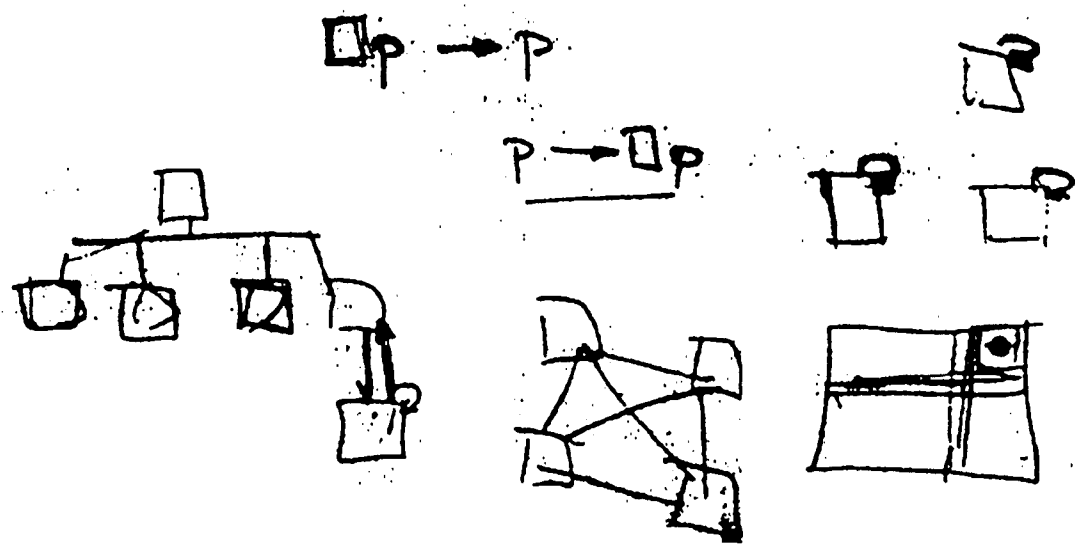
Subject 42



Subject 43



Subject 47



Appendix 6: MoLE code

This appendix contains the code for the program, MoLE.

View subclass: #UniverseView

```
instanceVariableNames: ""
classVariableNames: ""
poolDictionaries: ""
category: 'MLTool'!
```

!UniverseView methodsFor: 'controller accessing'!

defaultController

```
^UniverseController new.! !
```

!UniverseView methodsFor: 'displaying'!

displayOn: aGC

```
self model worlds
```

```
keysAndValuesDo:
```

```
[:key :value |
```

```
    | name length |
```

```
    name := key name.
```

```
    length := name size.
```

```
    length := length * 5 + 8.
```

```
    (Rectangle origin: value extent: length @ 24)
```

```
        displayStrokedOn: aGC.
```

```
    name displayString asComposedText displayOn: aGC at: value + (6 @ 7).
```

```
    key connections do:
```

```
[:each |
```

```
    | aPosition otherLength midA midB start finish |
```

```
    self model worlds keysAndValuesDo: [:world :position
```

```
    | world name = each
```

```
        ifTrue:
```

```
            [aPosition := position.
```

```
            otherLength := world name size * 5 + 8]].
```

```
            start := value.
```

```
            finish := aPosition + (otherLength @ 24).
```

```
            midA := value + (length / 2 @ 12).
```

```
            midB := aPosition + (otherLength / 2 @ 12).
```

```
            midB x > (midA x - (length / 4))
```

```
                ifTrue:
```

```
                    [start := start + (length / 2 @ 0).
```

```
                    finish := finish - (otherLength / 2 @ 0)].
```

```
            midB x > (midA x + (length / 4))
```

```
                ifTrue:
```

```
                    [start := start + (length / 2 @ 0).
```

```
                    finish := finish -
```

```
                        (otherLength / 2 @ 0)].
```

```
            midB y > (midA y - 8)
```

```
                ifTrue:
```

```
                    [start := start + (0 @ 12).
```

```
                    finish := finish - (0 @ 12)].
```

```
            midB y > (midA y + 8)
```

```
                ifTrue:
```

```

    [start := start + (0 @ 12).
    finish := finish - (0 @ 12)].
key name = each
ifFalse:
[| theta disArray lineLength diffX diffY |
diffX := finish x - start x.
diffY := finish y - start y.
lineLength := ((diffX **2) + (diffY **2)) sqrt.
lineLength := lineLength - 10.
diffX = 0
ifTrue: [diffY<0 ifTrue: [theta := 0]
ifFalse: [theta := 3.1419]]
ifFalse: [theta := (diffY / diffX) arcTan].
(LineSegment from: start to: finish)
    displayStrokedOn: aGC.
    disArray := Array new.
    disArray changeSizeTo: 3.
    disArray at: 1 put: finish.
disArray at: 2 put: finish - (10*(((theta+0.4) cos)@((theta+0.4) sin))).
disArray at: 3 put: finish - (10*(((theta-0.4) cos)@((theta-0.4) sin))).
    diffX<0 ifTrue: {
disArray at: 2 put: finish + (10*(((theta+0.4) cos)@((theta+0.4) sin))).
disArray at: 3 put: finish + (10*(((theta-0.4) cos)@((theta-0.4) sin))).
    diffX = 0 ifTrue: [diffY>0 ifTrue: {
disArray at: 2 put: finish - (5@5).
disArray at: 3 put: finish - ((-5)@5)]
    ifFalse: {
disArray at: 2 put: finish + (5@5).
disArray at: 3 put: finish + ((-5)@5)]}.
(Polyline vertices: disArray)
displayFilledOn: aGC]]! !

```

Controller subclass: #UniverseController

```
instanceVariableNames: 'world universeInterface dispatcher lastPosition
                        carrying icon offset '
```

```
classVariableNames: "
```

poolDictionaries: "

category: 'MLTool'!

!UniverseController methodsFor: 'accessing'!

carrying

^carrying!

carrying: aWorld

carrying:= aWorld!

icon

```
(self carrying) isNil ifTrue: [icon:= Image extent: 8@8 depth: 1 palette: MappedPalette
whiteBlack
```

bits: #(2r00000000

2r00000000

2r00000000

2r00000000

2r00000000

2r00000000

2r00000000

```
2r00000000| pad: 8.]
```

```
if False: ( |length temp |
```

$$\text{length} := (\text{carrying name size}) * 5 + 8.$$

```
temp := Pixmap extent: length@24.
```

temp graphicsContext displayRectangle:

(Rectangle origin: 0@0 extent: length@24).

```
icon := CachedImage on: (temp asImage)].
```

^icon!

placeNew

```
(self model worldToPlace isNil) ifTrue: {^false} ifFalse: {^true}.
```

universeInterface: anInterface

universeInterface := anInterface.!

worldDelete

```
(self model worldToDelete isNil) ifTrue: [^false] ifFalse: [^true].!
```

!UniverseController methodsFor: 'control defaults'!

controlActivity

```
| aGraphicsContext |
```

"Sense for the red button being pressed, and act accordingly."

```
aGraphicsContext := (self view) graphicsContext.
```

```
(self sensor redButtonPressed) ifTrue: {
```

```
(lastPosition = (self sensor cursorPoint)) ifTrue: [self doubleClick].
```

```
lastPosition := (self sensor cursorPoint).
```

```
self.getWorld.
```

```
(self icon) follow: [(self sensor cursorPoint) - offset]
```

```

        while: [self sensor redButtonPressed]
        on: aGraphicsContext.
    self replaceWorld.
    self model changed: #worlds].

^super controlActivity.!

controlInitialize

Cursor hand show.
    (model worldToPlace isNil) ifFalse: [Cursor crossHair show].
    (model worldToDelete isNil) ifFalse: [Cursor dropNotOk show].!

controlTerminate
    Cursor normal show.
    ^super controlTerminate.!

!UniverseController methodsFor: 'event dispatching'

getDispatcher
    ^dispatcher!

setDispatcher: anObject
    dispatcher:= anObject!

!UniverseController methodsFor: 'initialize'

initialize
    self setDispatcher: (UIDispatcher new doubleClick: [(self placeNew) ifFalse:
        [self getWorld.
        (carrying isNil) ifFalse: [universeInterface openWorld: carrying]]]).
    super initialize.!

!UniverseController methodsFor: 'private'

getWorld
    | whichPosition aWorld |
    whichPosition := self sensor cursorPoint.
    offset:=0.
    self placeNew
        ifTrue:
            [ | aNewWorld |
              aNewWorld := GridWorld new.
              aNewWorld universe: (self model).
              aNewWorld name: (aNewWorld getName).
              aNewWorld addDependent: universeInterface.
              self model worlds at: aNewWorld put: (whichPosition - (20@6)).
              aWorld := nil.
              self model worldToPlace: nil. Cursor hand show]
        ifFalse: [self model worlds keysAndValuesDo:
            [:key :value |
              | name length |
              name := key name.
              length := name size.
              length := length *5 + 8.
              ((Rectangle origin: value extent: length @ 24)
               containsPoint: whichPosition)
              ifTrue: [offset:= whichPosition-value.
                      aWorld := key]]].

    self carrying: aWorld.
    (carrying isNil) ifFalse: [(self worldDelete)

```

ifTrue:

```
[self model worlds keys do: [:each | each removeConnection: (carrying name)].  
self model without: carrying.  
carrying closeWindows.  
carrying := nil]].
```

self model worldToDelete: nil.

Cursor hand show!

replaceWorld

```
(carrying isNil) ifFalse: { | whichPosition |
```

```
whichPosition := (self sensor cursorPoint).
```

```
model worlds at: carrying put: (whichPosition-offset)].! !
```

```

Model subclass: #Universe
  instanceVariableNames: 'worlds worldToPlace worldToDelete worldToEvaluate
                          piece1 piece2 piece3 '
  classVariableNames: ''
  poolDictionaries: ''
  category: 'MLTool!'

```

```

!Universe methodsFor: 'accessing'!

```

```

piece1
  piece1 isNil ifTrue: (^String new)
    ifFalse: (^piece1 name)!

```

```

piece1: aPiece
  piece1:= aPiece.
  self changed: #piece1.!

```

```

piece2
  piece2 isNil ifTrue: (^String new)
    ifFalse: (^piece2 name)!

```

```

piece2: aPiece
  piece2:= aPiece.
  self changed: #piece2.!

```

```

piece3
  piece3 isNil ifTrue: (^String new)
    ifFalse: (^piece3 name)!

```

```

piece3: aPiece
  piece3:= aPiece.
  self changed: #piece3.!

```

```

without: anObject
  | tempDict |

  tempDict := Dictionary new.

```

```

  worlds keysAndValuesDo: [:key :value | (key = anObject) ifFalse:
    [tempDict at: key put: value]].
  worlds:= tempDict.!

```

```

worlds
  ^worlds!

```

```

worlds: aDictionary
  worlds := aDictionary.!

```

```

worldToDelete
  ^worldToDelete!

```

```

worldToDelete: aValue
  worldToDelete:= aValue.
  worldToPlace := nil.!

```

```

worldToEvaluate
  worldToEvaluate isNil ifTrue: (^String new)
    ifFalse: (^worldToEvaluate name)!

```

```

worldToEvaluate: aWorld
  worldToEvaluate:= aWorld.
  self changed: #worldToEvaluate.!

```

```

worldToPlace
  ^worldToPlace!

```

```

worldToPlace: aValue
  worldToPlace:= aValue.
  worldToDelete := nil.! !

```

```

!Universe methodsFor: 'evaluating'!

```

```

evaluate: aClaim

```

```

| temp |

```

```

Cursor wait show.
  GridWorld Example: nil.
  GridWorld HasWarned: false.
  temp := (worldToEvaluate evaluate: aClaim).
Cursor normal show.
  ^temp! !

```

```

!Universe methodsFor: 'initialize-release'!

```

```

initialize
  "creates an empty dictionary to store gridWorlds in"
  worlds:= Dictionary new.
  worldToPlace := nil.
  worldToDelete := nil.
  GridWorld GridSize: (6@6).! !

```

```

!Universe methodsFor: 'printing'!

```

```

printOn: aStream

```

```

| aList |

```

```

aStream nextPutAll: 'This Universe contains the following worlds: '.

```

```

aList := List new: self size.
worlds do: [:each | aList addLast: Character cr; addLast: each].

```

```

aStream nextPutAll: (aList printString).! !

```

```

!Universe methodsFor: 'updating'!

```

```

update: anAspect
  anAspect == #worlds ifTrue: [self changed].
  anAspect == #name ifTrue: [self changed: #worlds]!

```

```

update: anAspect with: aValue
  anAspect == #name ifTrue: [self changed: #name with: aValue].
  anAspect == #worlds ifTrue: [self changed: #worlds].! !

```

```

"-----"!

```

```

Universe class
  instanceVariableNames: ""!

```


!Universe class methodsFor: 'initialize-release'!

new
^super new initialize.! !

Cursor hand show.!

controlTerminate

Cursor normal show.

^super controlTerminate.!!

!GridController methodsFor: 'initialize'

initialize

```
(Blank == nil) ifTrue: [ | temp |  
    temp := Image extent: 8@8 depth: 1 palette: MappedPalette whiteBlack  
        bits: #[2r00000000  
                2r00000000  
                2r00000000  
                2r00000000  
                2r00000000  
                2r00000000  
                2r00000000] pad: 8.
```

Blank:= CachedImage on: temp].

icon:= Blank.

self setDispatcher: (UIDispatcher new doubleClick: [(self placeNew) ifFalse: [self getIcon.
 (carrying isNil) ifFalse: [gridWorldInterface editPiece: carrying]]]).

super initialize.!!

!GridController methodsFor: 'private'

getGridDropPosition

| scale origin whichPosition |

scale := (self model gridSize).
origin := centre - ((22*(scale x))@(22*(scale y))).

whichPosition := (((self sensor lastUpPoint) + (8@8) - origin)/44) truncated.
whichPosition := ((whichPosition x + 1)@(scale y - whichPosition y)).

^whichPosition.!

getGridPosition

| scale origin whichPosition |

scale := (self model gridSize).
origin := centre - ((22*(scale x))@(22*(scale y))).

whichPosition := (((self sensor cursorPoint) + (8@8) - origin)/44) truncated.
whichPosition := ((whichPosition x + 1)@(scale y - whichPosition y)).

^whichPosition.!

getIcon

| aShape whichPosition |

whichPosition:= (self getGridPosition).

(self placeNew) ifTrue: [aShape:= (self model pieceToPlace)]

```

                                ifFalse: [aShape:=nil.
model pieces keysAndValuesDo: [:key :value |
value = whichPosition ifTrue: [aShape:=key]]].

self carrying: aShape.
(aShape isNil) ifTrue: [self icon: (self blank)]
ifFalse: [(aShape isCircle) ifTrue: [self icon: (model circle)].
(aShape isSquare) ifTrue: [self icon: (model square)].
(aShape isTriangle) ifTrue: [self icon: (model tetrahedron)]]!.

replacePiece

(carrying isNil) ifFalse: [ | whichPosition |

whichPosition := (self getGridDropPosition).

self placeNew                ifTrue: [ (self model) addPiece: carrying at: whichPosition.
                                (self model) pieceToPlace: 'empty']
                                ifFalse: [ | anotherPosition |
                                anotherPosition := (model pieces at: carrying).
                                (anotherPosition = whichPosition) ifFalse: [
                                (self model) movePieceFrom: anotherPosition
                                to: whichPosition.]]!.!

!GridController methodsFor: 'event dispatching'

getDispatcher
    ^dispatcher!

setDispatcher: anObject
    dispatcher:= anObject! !

```

```

ApplicationModel subclass: #UniverseInterface
  instanceVariableNames: 'universe universeView gridX gridY chosenWorld testString object1
object2 object3 onlyCirclesMove system justDone testArray'
  classVariableNames: ""
  poolDictionaries: ""
  category: 'MLTool!'

```

!UniverseInterface methodsFor: 'initialize-release'!

initialize

```

universe := Universe new.
universeView := UniverseView new.
universe addDependent: self.
(self universeView) model: (self universe).
(self universeView) controller: UniverseController new.
(self universeView controller) universeInterface: self.
justDone := false.!!

```

!UniverseInterface methodsFor: 'actions'!

addAbove

! userHasAccepted success !

```

justDone ifTrue: [justDone := false. self clearSentences].

```

```

(universe worldToEvaluate = "") ifTrue: [^Dialog warn: 'Sentences can only be evaluated once
you"ve\chosen a world in which to evaluate them.' withCRs]

```

```

    ifFalse: [
      (testString value = "") ifTrue: [testString value: 'In ', (chosenWorld value) displayString, ' '].
      userHasAccepted:= (self openDialogInterface: #objectChoiceSpec).
      userHasAccepted ifTrue: [ 1 piece1Name piece2Name |
        piece1Name := universe piece1 displayString.
        piece2Name := universe piece2 displayString.
        testArray value: (testArray value, ' Above (', piece1Name, ', ', piece2Name, ')).
        testString value: testString value, piece1Name, ' is above ', piece2Name]
      ifFalse: [^self]].

```

```

success := (universe evaluate: (testArray value)).

```

```

(success isNil) ifFalse: [success ifTrue: [testString value: (testString value, ' - TRUE.').
testArray value: (testArray value, ' - TRUE.')]
ifFalse: [testString value: (testString value, ' - FALSE.').
testArray value: (testArray value, ' - FALSE.')]

```

```

self offerExample: success].

```

```

justDone :=true.!!

```

addBelow

! userHasAccepted success !

```

justDone ifTrue: [justDone := false. self clearSentences].

```

```

(universe worldToEvaluate = "") ifTrue: [^Dialog warn: 'Sentences can only be evaluated once
you"ve\chosen a world in which to evaluate them.' withCRs]

```

```

    ifFalse: [
      (testString value = "") ifTrue: [testString value: 'In ', (chosenWorld value) displayString, ' '].
      userHasAccepted:= (self openDialogInterface: #objectChoiceSpec).

```

```

userHasAccepted ifTrue: [ | piece1Name piece2Name |
    piece1Name := universe piece1 displayString.
    piece2Name := universe piece2 displayString.
    testArray value: (testArray value, ' Below (', piece1Name, ', ', piece2Name, ')').
    testString value: testString value, piece1Name, ' is below ', piece2Name]
ifFalse: [^self]].

success := (universe evaluate: (testArray value)).

(success isNil) ifFalse: [success ifTrue: [testString value: (testString value, ' - TRUE.').
    testArray value: (testArray value, ' - TRUE.')]
    ifFalse: [testString value: (testString value, ' - FALSE.').
        testArray value: (testArray value, ' - FALSE.')]

    self offerExample: success].

justDone :=true.

addBetween
    | userHasAccepted success |

    justDone ifTrue: [justDone := false. self clearSentences].

    universe worldToEvaluate = "
        ifTrue: [^Dialog warn: 'Sentences can only be evaluated once you've\chosen a world in
which to evaluate them.' withCRs]
        ifFalse:
            [testString value = " ifTrue: [testString value: 'In ', chosenWorld value displayString
, ''].

            userHasAccepted := self openDialogInterface: #objectChoiceSpec.
            userHasAccepted
                ifTrue:
                    [| piece1Name piece2Name piece3Name |
                        piece1Name := universe piece1 displayString.
                        piece2Name := universe piece2 displayString.
                        piece3Name := universe piece3 displayString.
                        testArray value: testArray value, ' Between
(' , piece1Name, ', ', piece2Name, ', ', piece3Name, ')'.
                        testString value: testString value, piece1Name, ' is between ',
piece2Name, ' and ', piece3Name]
                    ifFalse: [^self]].

            success := (universe evaluate: (testArray value)).

            (success isNil) ifFalse: [success ifTrue: [testString value: (testString value, ' - TRUE.').
                testArray value: (testArray value, ' - TRUE.')]
                ifFalse: [testString value: (testString value, ' - FALSE.').
                    testArray value: (testArray value, ' - FALSE.')]

                self offerExample: success].

            justDone :=true.

addCan

    justDone ifTrue: [justDone := false. self clearSentences].

    (universe worldToEvaluate = "") ifTrue: [^Dialog warn: 'Sentences can only be evaluated once
you've\chosen a world in which to evaluate them.' withCRs]
        ifFalse: [
            (testString value = "") ifTrue: [testString value: 'In ', (chosenWorld value) displayString, ' '].

```

```
testString value: testString value, 'it can be the case that '.
testArray value: (testArray value, 'C')!]
```

addLeftOf

```
| userHasAccepted success |
```

```
justDone ifTrue: [justDone := false. self clearSentences].
```

```
(universe worldToEvaluate = "") ifTrue: [^Dialog warn: 'Sentences can only be evaluated once
you've\chosen a world in which to evaluate them.' withCRs]
```

```
ifFalse: [
(testString value = "") ifTrue: [testString value: 'In ', (chosenWorld value) displayString, ' '.
userHasAccepted:= (self openDialogInterface: #objectChoiceSpec).
userHasAccepted ifTrue: [ | piece1Name piece2Name |
    piece1Name := universe piece1 displayString.
    piece2Name := universe piece2 displayString.
    testArray value: (testArray value, ' LeftOf (', piece1Name, ', ', piece2Name, ')).
    testString value: testString value, piece1Name, ' is left of ', piece2Name]
ifFalse: [^self]].
```

```
success := (universe evaluate: (testArray value)).
```

```
(success isNil) ifFalse: [success ifTrue: [testString value: (testString value, ' - TRUE.').
    testArray value: (testArray value, ' - TRUE.')]
ifFalse: [testString value: (testString value, ' - FALSE.').
    testArray value: (testArray value, ' - FALSE.').
```

```
self offerExample: success].
```

```
justDone :=true.!
```

addMust

```
justDone ifTrue: [justDone := false. self clearSentences].
```

```
(universe worldToEvaluate = "") ifTrue: [^Dialog warn: 'Sentences can only be evaluated once
you've\chosen a world in which to evaluate them.' withCRs]
```

```
ifFalse: [
(testString value = "") ifTrue: [testString value: 'In ', (chosenWorld value) displayString, ' '.
testString value: testString value, 'it must be the case that '.
testArray value: (testArray value, 'M')!]
```

addRightOf

```
| userHasAccepted success |
```

```
justDone ifTrue: [justDone := false. self clearSentences].
```

```
(universe worldToEvaluate = "") ifTrue: [^Dialog warn: 'Sentences can only be evaluated once
you've\chosen a world in which to evaluate them.' withCRs]
```

```
ifFalse: [
(testString value = "") ifTrue: [testString value: 'In ', (chosenWorld value) displayString, ' '.
userHasAccepted:= (self openDialogInterface: #objectChoiceSpec).
userHasAccepted ifTrue: [ | piece1Name piece2Name |
    piece1Name := universe piece1 displayString.
    piece2Name := universe piece2 displayString.
    testArray value: (testArray value, ' RightOf (', piece1Name, ', ', piece2Name, ')).
    testString value: testString value, piece1Name, ' is right of ', piece2Name]
ifFalse: [^self]].
```

```
success := (universe evaluate: (testArray value)).
```

```

(success isNil) ifFalse: [success ifTrue: [testString value: (testString value, ' - TRUE.').
    testArray value: (testArray value, ' - TRUE.')]
    ifFalse: [testString value: (testString value, ' - FALSE.').
        testArray value: (testArray value, ' - FALSE.')]}.

```

```

    self offerExample: success].

```

```

justDone :=true.

```

```

changeConnectionsFrom: aValue to: aNewValue

```

```

    universe worlds keys do: [:each |
        (each connections includes: aValue) ifTrue:
            [each removeConnection: aValue; addConnection: aNewValue.
                each dependents do:
                    [:aDependent |
                        (aDependent class = GridWorldInterface)
                        ifTrue: [aDependent connectionsList list: (each connections)]]].

```

```

clearSentences

```

```

    testString value: ".
    testArray value: "!.

```

```

deleteWorld

```

```

    universe worldToDelete: true.

```

```

getObject1

```

```

    | mb tempString aWorld |

```

```

    universe worlds keys do: [: each | each name = universe worldToEvaluate ifTrue:
        [aWorld:= each]].

```

```

    mb := MenuBuilder new.

```

```

    (self sort: (aWorld pieces keys)) do: [:each |
        mb addLabel: (each displayString) value: (each printString)].

```

```

    tempString := (mb startUp).

```

```

    (aWorld pieces keys) do: [:each | each printString = tempString ifTrue: [universe piece1:
each]].

```

```

getObject2

```

```

    | mb tempString aWorld |

```

```

    universe worlds keys do: [: each | each name = universe worldToEvaluate ifTrue:
        [aWorld:= each]].

```

```

    mb := MenuBuilder new.

```

```

    (self sort: (aWorld pieces keys)) do: [:each |
        mb addLabel: (each displayString) value: (each printString)].

```

```

    tempString := (mb startUp).

```

```

    (aWorld pieces keys) do: [:each | each printString = tempString ifTrue: [universe piece2:
each]].

```

```

getObject3

```

```

    | mb tempString aWorld |

```



```

universe worlds keys do: [:each | each name = universe worldToEvaluate ifTrue:
    [aWorld:= each]].

mb := MenuBuilder new.

(self sort: (aWorld pieces keys)) do: [:each |
    mb addLabel: (each displayString) value: (each printString)].

tempString := (mb startUp).

(aWorld pieces keys) do: [:each | each printString = tempString ifTrue: [universe piece3:
each]].!

getWorld
    | mb tempString |

mb := MenuBuilder new.

(self sort: (universe worlds keys)) do: [:each |
    mb addLabel: (each displayString) value: (each printString)].

tempString := (mb startUp).

(universe worlds keys) do: [:each | each name = tempString ifTrue: [universe
worldToEvaluate: each]].!

makeWorld
    universe worldToPlace: true.!

offerExample: aValue

    | userHasAccepted |
    (GridWorld Example isNil) ifFalse:
        [userHasAccepted := (Dialog confirm: 'This claim was ', aValue displayString, ' in ',
GridWorld Example name, '\Would you like to view this world?' withCRs).
        userHasAccepted ifTrue: [| aNewView |
            aNewView := GridWorldInterface new.
            aNewView gridWorld: (GridWorld Example).
            (GridWorld Example) addDependent: aNewView.
            aNewView addDependent: self.
            (aNewView gridViewView) model: (aNewView gridWorld).
            aNewView open]].!

openWorld: aWorld
    | aNewView |

aNewView := GridWorldInterface new.
aNewView gridWorld: aWorld.
aWorld addDependent: aNewView.
aNewView addDependent: self.
(aNewView gridViewView) model: (aNewView gridWorld).
aNewView open.!

reset
    | surety |

    (universe worlds isEmpty) ifFalse: {
        surety := (Dialog confirm: 'Are you sure you want to delete\all the worlds you have
created\and start again?' withCRs).
        surety ifTrue: [universe worlds keys do: [:each | each closeWindows].
            universe worlds: Dictionary new; piece1: nil; piece2: nil; piece3: nil.

```

```

universe worldToEvaluate: nil.
self clearSentences.
universe changed: #worlds.]]!

```

```

resizeGrid
| places |

```

```

places := universe worlds keys.
(places size = 0) ifTrue: [
    (gridX value>0) ifFalse: [gridX value: 1].
    (gridX value<7) ifFalse: [gridX value: 6].
    (gridY value>0) ifFalse: [gridY value: 1].
    (gridY value<7) ifFalse: [gridY value: 6].
    GridWorld GridSize: ((gridX value)@(gridY value))]
    ifFalse: [
        gridX value: GridWorld GridSize x.
        gridY value: GridWorld GridSize y.

```

```

    ^Dialog warn: 'You cannot resize a grid once worlds have been created.\Use the "Start
over" button to reset and begin again.' withCRs]!

```

```

restrictGridWorld
GridWorld OnlyCircles: (onlyCirclesMove value)!

```

```

setEmUp
self openInterface: #SetUpSpec.
self updateSystem.
self changed: #connections.!

```

```

sort: aSet
| changes aCollection |
changes := true.
aCollection := OrderedCollection new.
aSet do: [:each | aCollection add: each].

```

```

[changes] whileTrue: [| temp |
    changes:= false.
    1 to: (aCollection size-1) do: [:index |
        (aCollection at: index) > (aCollection at: index+1)
        ifTrue: {temp := aCollection at: index.
            aCollection at: index put: (aCollection at: index+1).
            aCollection at: (index+1) put: temp.
            changes := true}]].
    ^aCollection.! !

```

```

!UniverseInterface methodsFor: 'aspects'!

```

```

chosenWorld
^chosenWorld isNil
    ifTrue:
        [chosenWorld := (AspectAdaptor subject: universe sendsUpdates: true)
            forAspect: #worldToEvaluate]
    ifFalse:
        [chosenWorld].!

```

```

gridX
^gridX isNil
    ifTrue:
        [gridX := 6 asValue]
    ifFalse:
        [gridX]!

```

```

gridY
  ^gridY isNil
    ifTrue:
      [gridY := 6 asValue]
    ifFalse:
      [gridY]!

object1
  ^object1 isNil
    ifTrue:
      [object1 := (AspectAdaptor subject: universe sendsUpdates: true) forAspect: #piece1]
    ifFalse:
      [object1]!

object2
  ^object2 isNil
    ifTrue:
      [object2 := (AspectAdaptor subject: universe sendsUpdates: true) forAspect: #piece2]
    ifFalse:
      [object2]!

object3
  ^object3 isNil
    ifTrue:
      [object3 := (AspectAdaptor subject: universe sendsUpdates: true) forAspect: #piece3]
    ifFalse:
      [object3]!

onlyCirclesMove
  ^onlyCirclesMove isNil
    ifTrue:
      [onlyCirclesMove := true asValue]
    ifFalse:
      [onlyCirclesMove]!

system
  ^system isNil
    ifTrue:
      [system := #T asValue]
    ifFalse:
      [system]!

testArray
  ^testArray isNil
    ifTrue:
      [testArray := String new asValue]
    ifFalse:
      [testArray]!

testString
  ^testString isNil
    ifTrue:
      [testString := String new asValue]
    ifFalse:
      [testString]!

universe
  ^universe!

universeView

```

```

^universeView!

updateSystem
    GridWorld System: (system value)! !

!UniverseInterface methodsFor: 'updating'!

update: anAspect
    anAspect == #name ifTrue: [self universeView invalidate].
    anAspect == #worlds ifTrue: [self universeView invalidate].
    anAspect == #connections ifTrue: [self universeView invalidate].!

update: anAspect with: aValue
    anAspect == #name ifTrue: [self changeConnectionsFrom: (aValue at: 1) to: (aValue at: 2).
        self universeView invalidate].
    anAspect == #worlds ifTrue: [self universeView invalidate].
    anAspect == #connections ifTrue: [self universeView invalidate].
    anAspect == #worldToEvaluate ifTrue: [self clearSentences].! !
"-----"!

UniverseInterface class
    instanceVariableNames: ''

!UniverseInterface class methodsFor: 'interface specs'!

objectChoiceSpec
    "UIPainter new openOnClass: self andSelector: #objectChoiceSpec"

<resource: #canvas>
^#(#FullSpec
    #window:
        #(#WindowSpec
            #label: ''
            #min: #(#Point 238 191 )
            #max: #(#Point 238 191 )
            #bounds: #(#Rectangle 230 161 468 352 ) )
        #component:
            #(#SpecCollection
                #collection: #(
                    #(#LabelSpec
                        #layout: #(#Point 30 0 )
                        #label: 'Piece to test:'
                        #style: #small )
                    #(#LabelSpec
                        #layout: #(#LayoutOrigin 0 0.12605 0 0.256544 )
                        #label: 'Compare it with:'
                        #style: #small )
                    #(#LabelSpec
                        #layout: #(#LayoutOrigin 0 0.100917 0 0.531646 )
                        #label: '...and: (only needed for "between")'
                        #style: #small )
                    #(#ActionButtonSpec
                        #layout: #(#LayoutFrame 0 0.0229358 0 0.101266 0 0.114679 0 0.227848 )
                        #model: #getObject1
                        #label: #ImageLabel
                        #hasCharacterOrientedLabel: false
                        #defaultable: true )
                    #(#ActionButtonSpec
                        #layout: #(#LayoutFrame 0 0.0229358 0 0.35443 0 0.114679 0 0.481013 )
                        #model: #getObject2

```

```

        #label: #ImageLabel
        #hasCharacterOrientedLabel: false
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.0229358 0 0.632911 0 0.114679 0 0.759494 )
        #model: #getObject3
        #label: #ImageLabel
        #hasCharacterOrientedLabel: false
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.168067 0 0.842932 0 0.378151 0 0.968586 )
        #model: #accept
        #label: 'Ok'
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.60084 0 0.842932 0 0.810924 0 0.968586 )
        #model: #cancel
        #label: 'Cancel'
        #defaultable: true )
    #(#InputFieldSpec
        #layout: #(#LayoutFrame 0 0.123853 0 0.107595 0 0.894495 0 0.227848 )
        #model: #object1
        #isReadOnly: true )
    #(#InputFieldSpec
        #layout: #(#LayoutFrame 0 0.123853 0 0.360759 0 0.894495 0 0.481013 )
        #model: #object2
        #isReadOnly: true )
    #(#InputFieldSpec
        #layout: #(#LayoutFrame 0 0.123853 0 0.63924 0 0.894495 0 0.759494 )
        #model: #object3
        #isReadOnly: true ) ) ) )!

```

SetUpSpec

"UIPainter new openOnClass: self andSelector: #SetUpSpec"

```

<resource: #canvas>
^#(#FullSpec
    #window:
    #(#WindowSpec
        #label: 'Set up initial conditions'
        #min: #(#Point 217 158 )
        #max: #(#Point 217 158 )
        #bounds: #(#Rectangle 476 519 693 677 ) )
    #component:
    #(#SpecCollection
        #collection: #(
            #(#LabelSpec
                #layout: #(#Point 7 13 )
                #label: 'Grid size is:' )
            #(#LabelSpec
                #layout: #(#Point 121 13 )
                #label: 'by:' )
            #(#InputFieldSpec
                #layout: #(#Rectangle 88 12 113 35 )
                #model: #gridX
                #callbacksSpec:
                #(#UIEventCallbackSubSpec
                    #valueChangeSelector: #resizeGrid )
                #type: #number )
            #(#InputFieldSpec
                #layout: #(#Rectangle 154 12 179 35 )

```

```

#model: #gridY
#callbacksSpec:
#(#UIEventCallbackSubSpec
    #valueChangeSelector: #resizeGrid )
#type: #number )
#(#CheckBoxSpec
    #layout: #(#Point 7 42 )
    #model: #onlyCirclesMove
    #callbacksSpec:
    #(#UIEventCallbackSubSpec
        #valueChangeSelector: #restrictGridWorld )
    #label: 'Only spheres can move' )
#(#ActionButtonSpec
    #layout: #(#Rectangle 49 123 178 150 )
    #model: #closeRequest
    #label: 'Finished setting up'
    #defaultable: true )
#(#LabelSpec
    #layout: #(#Point 61 67 )
    #label: 'Modal system' )
#(#RadioButtonSpec
    #layout: #(#Point 7 88 )
    #model: #system
    #callbacksSpec:
    #(#UIEventCallbackSubSpec
        #valueChangeSelector: #updateSystem )
    #label: 'T'
    #select: #T )
#(#RadioButtonSpec
    #layout: #(#Point 86 88 )
    #model: #system
    #callbacksSpec:
    #(#UIEventCallbackSubSpec
        #valueChangeSelector: #updateSystem )
    #label: 'S4'
    #select: #S4 )
#(#RadioButtonSpec
    #layout: #(#Point 174 88 )
    #model: #system
    #callbacksSpec:
    #(#UIEventCallbackSubSpec
        #valueChangeSelector: #updateSystem )
    #label: 'S5'
    #select: #S5 ) ) )

```

windowSpec

```
"UIPainter new openOnClass: self andSelector: #windowSpec"
```

```
<resource: #canvas>
```

```
^#(#FullSpec
```

```
    #window:
```

```
    #(#WindowSpec
```

```
        #label: 'Universe'
```

```
        #min: #(#Point 40 20 )
```

```
        #bounds: #(#Rectangle 7 40 388 394 )
```

```
        #menu: #menuBar )
```

```
    #component:
```

```
    #(#SpecCollection
```

```
        #collection: #(
```

```
            #(#ArbitraryComponentSpec
```

```
                #layout: #(#LayoutFrame 0 0.00829876 0 0.00221729 0 0.991701 0 0.603104 )
```

```

#flags: 11
#colors:
#(#LookAndFeel
  #setForegroundColor: #(#ColorValue #black )
  #setBackgroundColor: #(#ColorValue 6553 6553 6553 )
  #setSelectionForegroundColor: #(#ColorValue #black )
  #setSelectionBackgroundColor: #(#ColorValue 6553 6553 6553 ) )
#component: #universeView )
#(#ActionButtonSpec
  #layout: #(#LayoutFrame 0 0.553942 0 0.607539 0 0.771784 0 0.662971 )
  #model: #makeWorld
  #label: 'Add a new world'
  #style: #small
  #defaultable: true )
#(#ActionButtonSpec
  #layout: #(#LayoutFrame 0 0.773859 0 0.607539 0 0.991701 0 0.662971 )
  #model: #deleteWorld
  #label: 'Delete a world'
  #style: #small
  #defaultable: true )
#(#ActionButtonSpec
  #layout: #(#LayoutFrame 0 0.00829876 0 0.607539 0 0.307054 0 0.662971 )
  #model: #setEmUp
  #label: 'Set Up Initial Conditions'
  #style: #small
  #defaultable: true )
#(#DividerSpec
  #layout: #(#LayoutFrame 0 0.0954357 0 0.678492 0 0.906639 0 0.687361 ) )
#(#LabelSpec
  #layout: #(#LayoutOrigin 0 0.00787402 0 0.70339 )
  #label: 'Evaluate the truth of sentences in:'
  #style: #small )
#(#ActionButtonSpec
  #layout: #(#LayoutFrame 0 0.519685 0 0.69209 0 0.577428 0 0.751412 )
  #model: #getWorld
  #label: #ImageLabel
  #hasCharacterOrientedLabel: false
  #style: #small
  #defaultable: true )
#(#InputFieldSpec
  #layout: #(#LayoutFrame 0 0.577428 0 0.694915 0 0.845144 0 0.748588 )
  #model: #chosenWorld
  #style: #small
  #isReadOnly: true )
#(#ActionButtonSpec
  #layout: #(#LayoutFrame 0 0.643154 0 0.760532 0 0.744813 0 0.813747 )
  #model: #addMust
  #label: 'Must'
  #style: #small
  #defaultable: true )
#(#ActionButtonSpec
  #layout: #(#LayoutFrame 0 0.643154 0 0.818182 0 0.744813 0 0.873614 )
  #model: #addCan
  #label: 'Can'
  #style: #small
  #defaultable: true )
#(#ActionButtonSpec
  #layout: #(#LayoutFrame 0 0.748963 0 0.760532 0 0.860996 0 0.813747 )
  #model: #addAbove
  #label: 'is above'
  #style: #small

```

```

        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.748963 0 0.818182 0 0.860996 0 0.873614 )
        #model: #addBelow
        #label: 'is below'
        #style: #small
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.865145 0 0.760532 0 0.991701 0 0.813747 )
        #model: #addLeftOf
        #label: 'is left of'
        #style: #small
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.865145 0 0.818182 0 0.991701 0 0.873614 )
        #model: #addRightOf
        #label: 'is right of'
        #style: #small
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.848548 0 0.878049 0 0.991701 0 0.931264 )
        #model: #addBetween
        #label: 'is between'
        #style: #small
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.643154 0 0.878049 0 0.844398 0 0.931264 )
        #model: #clearSentences
        #label: 'Clear sentences'
        #style: #small
        #defaultable: true )
    #(#TextEditorSpec
        #layout: #(#LayoutFrame 0 0.00829876 0 0.758315 0 0.634855 0 0.931264 )
        #model: #testString
        #style: #small
        #isReadOnly: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.309129 0 0.607539 0 0.551867 0 0.662971 )
        #model: #reset
        #label: 'Reset & start again'
        #style: #small
        #defaultable: true )
    #(#InputFieldSpec
        #layout: #(#LayoutFrame 0 0.00829876 0 0.940133 0 0.991701 0 0.993348 )
        #model: #testArray
        #style: #small
        #isReadOnly: true
        #type: #string ) ) ) !

```

!UniverseInterface class methodsFor: 'resources'!

ImageLabel

! anImage !

anImage:= Image extent: 16@16 depth: 1 palette: (MappedPalette with: (ColorValue lightGray)
with: (ColorValue black))

```

bits: #|2r00000000 2r00000000
      2r00000000 2r00000000
      2r00000000 2r00000000
      2r00000000 2r00000000

```



```

2r10000000 2r00000001
2r11000000 2r00000011
2r01100000 2r00000110
2r00110000 2r00001100
2r00011000 2r00011000
2r00001100 2r00110000
2r00000110 2r01100000
2r00000011 2r11000000
2r00000001 2r10000000
2r00000000 2r00000000
2r00000000 2r00000000
2r00000000 2r00000000] pad: 16.

```

^anImage!

menuBar

"UIMenuEditor new openOnClass: self andSelector: #menuBar"

<resource: #menu>

^#(#Menu #(

 #(#MenuItem

 #rawLabel: 'File'

 #submenu: #(#Menu #(

 #(#MenuItem

 #rawLabel: 'Close')) #(1) nil))) #(1) nil) decodeAsLiteralArray! !

```

View subclass: #GridWorldView
  instanceVariableNames: "
  classVariableNames: "
  poolDictionaries: "
  category: 'MLTool!'

```

```
!GridWorldView methodsFor: 'controller accessing'
```

```
defaultController
  ^GridController new!!
```

```
!GridWorldView methodsFor: 'displaying'
```

```
displayOn: aGraphicsContext
```

```
  | worldX worldY centre topLeftCorner topRCorner bottomLCorner bottomRCorner |
```

```

  worldX := (model gridSize) x.
  worldY := (model gridSize) y.
  centre:= (((7*aGraphicsContext medium width)/10) @
    ((9*aGraphicsContext medium height)/10))/2.
  controller centre: centre.

```

```

  topLeftCorner := centre-((22*worldX)@(22*worldY)).
  bottomRCorner := centre+((22*worldX)@(22*worldY)).
  topRCorner := ((bottomRCorner x)@(topLCorner y)).
  bottomLCorner := ((topLCorner x)@(bottomRCorner y)).

```

```
  1 to: (worldX+1) do: [:index | | xStart xStop |
```

```

    xStart:= topLeftCorner + ((index-1)*(topRCorner-topLCorner)/(worldX)).
    xStop:= bottomLCorner + ((index-1)*(bottomRCorner-bottomLCorner)/(worldX)).
    (LineSegment from: xStart to: xStop) displayStrokedOn: aGraphicsContext].

```

```
  1 to: (worldY+1) do: [:yIndex | | start stop |
```

```

    start:=topLCorner + ((yIndex-1)*(bottomLCorner-topLCorner)/(worldY)).
    stop:=topRCorner + ((yIndex-1)*(bottomRCorner-topRCorner)/(worldY)).

```

```
    (LineSegment from: start to: stop) displayStrokedOn: aGraphicsContext].
```

```

(model pieces) keysAndValuesDo:
  [:key :value | | placeIt length |

```

```
    placeIt := topLeftCorner + (((value x)*44)@(((worldY +1 - value y)*44))) -(22@22).
```

```

    length := ((key name) size).
    length := length*4.

```

```

    (key name) displayString asComposedText displayOn: aGraphicsContext
    at: (placeIt-((length/2)@(-11))).

```

```

    ((key shape)='circle') ifTrue: [ (model circle) displayOn: aGraphicsContext
                                     at: (placeIt-(15@20))]

```

```

    ((key shape)='cube') ifTrue: [ (model square) displayOn: aGraphicsContext
                                   at: (placeIt-(15@20))].

```

```

    ((key shape)='tetrahedron') ifTrue: [(model tetrahedron)
                                          displayOn: aGraphicsContext
                                          at: (placeIt-(15@20))]]! !

```

```

ApplicationModel subclass: #GridWorldInterface
    instanceVariableNames: 'gridWorld gridWorldView gridNameHolder editNameHolder
editPiece editShapeHolder connectionsList '
    classVariableNames: ''
    poolDictionaries: ''
    category: 'MLTool'

```

!GridWorldInterface methodsFor: 'actions'

```

addLink
    | mb tempString alreadyThere |

    mb := MenuBuilder new.
    alreadyThere := false.

    (gridWorld sort: (gridWorld universe worlds keys)) do: [:each |
        mb addLabel: (each displayString) value: (each displayString)].

    tempString := (mb startUp).

    gridWorld connections do: [:each | each = tempString ifTrue: [alreadyThere := true]].
    tempString = 0 ifTrue: [alreadyThere := true].

    alreadyThere ifFalse: [gridWorld addConnection: tempString.
        self connectionsList list: (gridWorld connections).
        self changed: #connections].!

```

changeShape
 "Creates a dialogue window allowing the user to edit a piece on the grid world"

```

(editShapeHolder value) = #Circle ifTrue: [editPiece beCircle].
(editShapeHolder value) = #Triangle ifTrue: [editPiece beTriangle].
(editShapeHolder value) = #Square ifTrue: [editPiece beSquare].!

```

```

clearBoard
    gridWorld pieces: Dictionary new.
    gridWorld changed: #pieces.!

```

```

closeWindow
    Cursor normal show.
    self closeRequest.!

```

```

copyGrid
    | mb tempString |

    mb := MenuBuilder new.

    (gridWorld universe worlds keys) do: [:each |
        mb addLabel: (each displayString) value: (each displayString)].

    tempString := (mb startUp).

    (gridWorld universe worlds keys) do: [:each | (each name) = tempString ifTrue:
        [gridWorld pieces: (each pieces)]].
    gridWorld changed: #pieces.!

```

```

editPiece: aPiece
    "Creates a dialogue window allowing the user to edit a piece on the grid world"
    editPiece := aPiece.

```

```

editNameHolder := (aPiece name) asValue.

(aPiece shape = 'circle') ifTrue: [editShapeHolder := #Circle asValue].
(aPiece shape = 'tetrahedron') ifTrue: [editShapeHolder := #Triangle asValue].
(aPiece shape = 'cube') ifTrue: [editShapeHolder := #Square asValue].

self openInterface: #dialogEditSpec.

enumerate
    gridWorld generateWorlds: 1.
    self connectionsList list: (gridWorld connections).
    self changed: #connections.

enumerate2
    gridWorld generateWorlds: 2.
    self connectionsList list: (gridWorld connections).
    self changed: #connections.

enumerate3
    gridWorld generateWorlds: 3.
    self connectionsList list: (gridWorld connections).
    self changed: #connections.

newCircle
    | temp |

    temp := Piece new.
    temp beCircle.
    temp name: 'piece ', ((gridWorld pieces size+1) displayString).
    (self gridWorld) pieceToPlace: temp.

newSquare
    | temp |

    temp := Piece new.
    temp name: 'piece ', ((gridWorld pieces size+1) displayString).
    (self gridWorld) pieceToPlace: temp.

newTetrahedron
    | temp |

    temp := Piece new.
    temp beTriangle.
    temp name: 'piece ', ((gridWorld pieces size+1) displayString).
    (self gridWorld) pieceToPlace: temp.

openLink
    (connectionsList selection isNil) ifFalse: [ | aNewView aWorld |

        (gridWorld universe worlds keys) do: [: each |
            (each name) = ((self connectionsList) selection) ifTrue: [aWorld:=each]].
        aNewView := GridWorldInterface new.
        aNewView gridWorld: aWorld.
        aWorld addDependent: aNewView.
        (aNewView gridWorldView) model: (aNewView gridWorld).
        aNewView open.

removeLink
    (connectionsList selection isNil) ifFalse: [ | tempCollection |
        tempCollection := OrderedCollection new.

```

```

gridWorld connections do: [: each |
    (each displayString) = (self connectionsList selection)
        ifFalse: [tempCollection add: each]].
gridWorld connections: tempCollection.
self connectionsList list: (gridWorld connections).
self changed: #connections!

removePiece
    (gridWorld universe worlds keys) do: [: each | each without: editPiece].
self closeRequest!

renamePiece
    editPiece name: (editNameHolder value)! !

!GridWorldInterface methodsFor: 'initialize-release'

initialize
    "Make a new world for the interface to manipulate"

    self gridWorld: GridWorld new.
    gridWorld addDependent: self.
    gridWorldView := GridWorldView new.
    gridWorldView controller: GridController new.
    (gridWorldView controller) gridWorldInterface: self.
    gridWorldView model: gridWorld.! !

!GridWorldInterface methodsFor: 'aspects'!

connectionsList
    ^connectionsList isNil
        ifTrue:
            [connectionsList := SelectionInList with: (gridWorld connections)]
        ifFalse:
            [connectionsList]!

editNameHolder
    ^editNameHolder isNil
        ifTrue:
            [editNameHolder := " asValue]
        ifFalse:
            [editNameHolder]!

editShapeHolder
    ^editShapeHolder isNil
        ifTrue:
            [editShapeHolder := nil asValue]
        ifFalse:
            [editShapeHolder]!

gridNameHolder
    ^gridNameHolder isNil
        ifTrue:
            [gridNameHolder :=
                (AspectAdaptor subject: gridWorld sendsUpdates: true)
                forAspect: #name]
        ifFalse:
            [gridNameHolder]!

gridWorld
    ^gridWorld!

```

```
gridWorld: aWorld
  gridWorld := aWorld.!
```

```
gridWorldView
  ^gridWorldView! !
```

```
!GridWorldInterface methodsFor: 'updating'!
```

```
update: anAspect
  anAspect == #shape ifTrue: [ gridWorldView changed ].! !
"-- -----"
```

```
GridWorldInterface class
  instanceVariableNames: ""
```

```
!GridWorldInterface class methodsFor: 'interface specs'!
```

```
dialogEditSpec
  "UIPainter new openOnClass: self andSelector: #dialogEditSpec"

  <resource: #canvas>
  ^#(#FullSpec
    #window:
      #(#WindowSpec
        #label: 'Editing a Piece'
        #min: #(#Point 235 151 )
        #max: #(#Point 235 151 )
        #bounds: #(#Rectangle 181 131 416 282 ) )
    #component:
      #(#SpecCollection
        #collection: #(
          #(#RadioButtonSpec
            #layout: #(#LayoutOrigin 0 0.394834 0 0.306122 )
            #model: #editShapeHolder
            #label: 'Tetrahedron'
            #select: #Triangle )
          #(#InputFieldSpec
            #layout: #(#LayoutFrame 0 0.313653 0 0.0357143 0 0.797048 0 0.173469 )
            #model: #editNameHolder )
          #(#ActionButtonSpec
            #layout: #(#LayoutFrame 0 0.236162 0 0.760204 0 0.719557 0 0.897959 )
            #model: #closeRequest
            #label: 'Finished editing'
            #isDefault: false
            #defaultable: true )
          #(#ActionButtonSpec
            #layout: #(#LayoutFrame 0 0.833948 0 0.535714 0 0.97786 0 0.653061 )
            #model: #removePiece
            #label: 'Ok'
            #defaultable: true )
          #(#ActionButtonSpec
            #layout: #(#LayoutFrame 0 0.833948 0 0.30102 0 0.97786 0 0.418367 )
            #model: #changeShape
            #label: 'Ok'
            #defaultable: true )
          #(#LabelSpec
            #layout: #(#LayoutOrigin 0 0.0184502 0 0.306122 )
            #label: 'Reshape as:' )
          #(#LabelSpec
            #layout: #(#LayoutOrigin 0 0.0184502 0 0.561224 )
```

```

        #label: 'Delete this piece.....' )
    #(#LabelSpec
        #layout: #(#LayoutOrigin 0 0.0184502 0 0.0510204 )
        #label: 'Rename as:' )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.833948 0 0.0408163 0 0.97786 0 0.158163 )
        #model: #renamePiece
        #label: 'Ok'
        #isDefault: true
        #defaultable: true )
    #(#RadioButtonSpec
        #layout: #(#LayoutOrigin 0 0.394834 0 0.193878 )
        #model: #editShapeHolder
        #label: 'Circle'
        #select: #Circle )
    #(#RadioButtonSpec
        #layout: #(#LayoutOrigin 0 0.394834 0 0.418367 )
        #model: #editShapeHolder
        #label: 'Cube'
        #select: #Square ) ) )!

```

windowSpec

"UIPainter new openOnClass: self andSelector: #windowSpec"

```

<resource: #canvas>
^#(#FullSpec
    #window:
    #(#WindowSpec
        #label: 'Grid World'
        #min: #(#Point 40 20 )
        #bounds: #(#Rectangle 95 117 491 432 )
        #flags: 4
        #menu: #menuBar )
    #component:
    #(#SpecCollection
        #collection: #(
            #(#ArbitraryComponentSpec
                #layout: #(#LayoutFrame 0 0.0020284 0 0.00290698 0 0.711967 0 0.982558)
                #component: #gridWorldView )
            #(#SequenceViewSpec
                #layout: #(#LayoutFrame 0 0.728195 0 0.488372 0 0.989858 0 0.787791 )
                #colors:
                #(#LookPreferences
                    #setForegroundColor: #(#ColorValue #black )
                    #setBackgroundColor: #(#ColorValue 6553 6553 6553 )
                    #setSelectionForegroundColor: #(#ColorValue #black )
                    #setSelectionBackgroundColor: #(#ColorValue 4915 4915 4915 ) )
                #model: #connectionsList
                #callbacksSpec:
                #(#UIEventCallbackSubSpec
                    #valueChangeSelector: #connectionsList )
                #multipleSelections: false
                #style: #small
                #selectionType: #highlight )
            #(#CompositeSpecCollection
                #collection: #(
                    #(#RadioButtonSpec
                        #layout: #(#Point 0 0 )
                        #label: 'Square' )
                    #(#RadioButtonSpec
                        #layout: #(#Point 0 43 )

```

```

        #label: 'Triangle' ) )
    #compositeSpec:
    #(#CompositeSpec
        #layout: #(#Rectangle 140 -161 208 -98 ) ) )
    #(#InputFieldSpec
        #layout: #(#LayoutFrame 0 0.72796 0 0.0541516 0 0.989924 0 0.115523 )
        #model: #gridNameHolder
        #callbacksSpec:
        #(#UIEventCallbackSubSpec
            #focusOutSelector: #gridNameHolder )
        #style: #small )
    #(#LabelSpec
        #layout: #(#LayoutOrigin 0 0.720403 0 0 )
        #label: 'This world is called:'
        #style: #small )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.717884 0 0.256318 0 0.808564 0 0.389892 )
        #model: #newCircle
        #label: #circleLabel
        #hasCharacterOrientedLabel: false
        #style: #small
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.813602 0 0.256318 0 0.904282 0 0.389892 )
        #model: #newTetrahedron
        #label: #tetrahedronLabel
        #hasCharacterOrientedLabel: false
        #style: #small
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.906801 0 0.256318 0 0.997481 0 0.389892 )
        #model: #newSquare
        #label: #squareLabel
        #hasCharacterOrientedLabel: false
        #style: #small
        #defaultable: true )
    #(#LabelSpec
        #layout: #(#LayoutOrigin 0 0.740554 0 0.151625 )
        #label: 'Click on a button'
        #style: #small )
    #(#LabelSpec
        #layout: #(#LayoutOrigin 0 0.770781 0 0.393502 )
        #label: 'This world is'
        #style: #small )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.726166 0 0.790698 0 0.989858 0 0.851744 )
        #model: #openLink
        #label: 'Open selected world'
        #style: #small
        #defaultable: true )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.726166 0 0.857558 0 0.989858 0 0.918605 )
        #model: #addLink
        #label: 'Add new connection'
        #style: #small
        #defaultable: true )
    #(#LabelSpec
        #layout: #(#LayoutOrigin 0 0.758186 0 0.433213 )
        #label: 'connected to:'
        #style: #small )
    #(#GroupBoxSpec

```



```

        #layout: #(#LayoutFrame 0 0.712846 0 0.148014 0 1.0 0 0.400722 )
        #style: #small )
    #(#LabelSpec
        #layout: #(#LayoutOrigin 0 0.753149 0 0.191336 )
        #label: 'to add a piece:'
        #style: #small )
    #(#ActionButtonSpec
        #layout: #(#LayoutFrame 0 0.726166 0 0.924419 0 0.989858 0 0.985465 )
        #model: #removeLink
        #label: 'Remove a connection'
        #style: #small
        #defaultable: true ) ) ) ! !

```

!GridWorldInterface class methodsFor: 'resources'!

```

circleLabel
    ^GridWorld circleImage.!

```

```

menuBar
    "MenuEditor new openOnClass: self andSelector: #menuBar"

```

```

<resource: #menu>
^#(#Menu #(
    #(#MenuItem
        #rawLabel: 'File'
        #submenu: #(#Menu #(
            #(#MenuItem
                #rawLabel: 'Close'
                #value: #closeWindow ) ) #(1 ) nil ) )
    #(#MenuItem
        #rawLabel: 'Edit'
        #submenu: #(#Menu #(
            #(#MenuItem
                #rawLabel: 'Move pieces and create worlds'
                #value: #enumerate )
            #(#MenuItem
                #rawLabel: 'Copy pieces in from...'
                #value: #copyGrid )
            #(#MenuItem
                #rawLabel: 'Wipe board clean'
                #value: #clearBoard )
            #(#MenuItem
                #rawLabel: 'Move pieces twice'
                #value: #enumerate2 )
            #(#MenuItem
                #rawLabel: 'Move pieces three times'
                #value: #enumerate3 ) ) #(3 2 ) nil ) ) ) #(2 ) nil )

```

decodeAsLiteralArray!

```

squareLabel
    ^GridWorld squareImage.!

```

```

tetrahedronLabel
    ^GridWorld tetrahedronImage.! !

```

!GridWorldInterface class methodsFor: 'instance creation'!

```

new
    ^super new initialize! !

```

```

Model subclass: #GridWorld
  instanceVariableNames: 'pieces name connections pieceToPlace universe hasSpawned '
  classVariableNames: 'CircleImage Example GridSize HasWarned OnlyCircles SquareImage
System TetrahedronImage '
  poolDictionaries: ''
  category: 'MLTool!'

```

```
!GridWorld methodsFor: 'accessing'!
```

```

addConnection: aConnection
  | tempCollection |

  tempCollection := OrderedCollection with: aConnection.
  connections do: [:each |
    tempCollection add: each].
  self connections: tempCollection.!

```

```

addPiece: aPiece at: aPosition
  (self isAnythingAt: aPosition)
    ifFalse: [pieces at: (aPiece) put: aPosition.].
  self changed: #pieces.
  aPiece addDependent: self.!

```

```

circle
  ^CircleImage!

```

```

circle: anImage
  CircleImage:= CachedImage on: anImage.!

```

```

connections
  ^connections!

```

```

connections: anOrderedCollection
  connections:= (self sort: anOrderedCollection).!

```

```

generateWorlds
  | index collection sameWorld userHasAccepted whereItsGoing |

```

```

userHasAccepted := false.
index := 0.

```

```

hasSpawned ifTrue: [userHasAccepted := (Dialog confirm: 'This world has already been used to
derive other possible worlds.\Unless you"ve deleted these, you will create duplicates, which
are\sure to confuse matters. Are you sure you want to proceed?' withCRs)].

```

```

userHasAccepted ifTrue: [hasSpawned := false].

```

```

hasSpawned ifFalse: [ |gathered |

```

```

gathered := OrderedCollection new.

```

```

  hasSpawned := true.

```

```

  (GridWorld OnlyCircles) ifTrue: [
    collection := Dictionary new.
    pieces keysAndValuesDo: [: aPiece :aValue |
      aPiece isCircle ifTrue: [collection at: aPiece put: aValue]]]
    ifFalse: [
      collection := pieces].

```

```

sameWorld := GridWorld new.
sameWorld universe: self universe.
sameWorld name: (self getName).
self pieces keysAndValuesDo: [:key :value | sameWorld pieces at: key put: value.
    key addDependent: sameWorld].
sameWorld addDependent: universe.
whereItsGoing := ((universe worlds at: self) + (0@((index+1)*50))).
[universe worlds values includes: whereItsGoing]
    whileTrue: [whereItsGoing:=whereItsGoing + (10@10)].
universe worlds at: sameWorld put: whereItsGoing.
gathered add: sameWorld.
self addConnection: sameWorld name.

collection keysAndValuesDo: [: aPiece :aValue | | tempPosition worldPosition |
    index := index +1.
    tempPosition := aValue + (0@1).
    (tempPosition y < (1+GridSize y)) ifTrue: [(self isAnythingAt: tempPosition) ifFalse:
        [| tempWorld |
            tempWorld := GridWorld new.
            tempWorld universe: self universe.
            tempWorld name: (self getName).
            self pieces keysAndValuesDo: [:key :value | tempWorld pieces at: key put: value.
                key addDependent: tempWorld].
            tempWorld addDependent: universe.
            tempWorld movePieceFrom: aValue to: tempPosition.
            worldPosition := (universe worlds at: self).
            whereItsGoing := (worldPosition + ((-100)@(index *50))).
            (whereItsGoing x < 0) ifTrue: [whereItsGoing x: ((whereItsGoing x) abs).
                whereItsGoing y: (whereItsGoing y +35)].
            [universe worlds values includes: whereItsGoing]
                whileTrue: [whereItsGoing:=whereItsGoing + (10@10)].
            universe worlds at: tempWorld put: whereItsGoing.
            gathered add: tempWorld.
            self addConnection: tempWorld name]].
    tempPosition := aValue - (0@1).
    (tempPosition y > 0) ifTrue: [(self isAnythingAt: tempPosition) ifFalse:
        [| tempWorld |
            tempWorld := GridWorld new.
            tempWorld universe: self universe.
            tempWorld name: (self getName).
            self pieces keysAndValuesDo: [:key :value | tempWorld pieces at: key put: value.
                key addDependent: tempWorld].
            tempWorld addDependent: universe.
            tempWorld movePieceFrom: aValue to: tempPosition.
            worldPosition := (universe worlds at: self).
            whereItsGoing := (worldPosition + ((-50)@(index *50))).
            (whereItsGoing x < 0) ifTrue: [whereItsGoing x: ((whereItsGoing x) abs).
                whereItsGoing y: (whereItsGoing y +35)].
            [universe worlds values includes: whereItsGoing]
                whileTrue: [whereItsGoing:=whereItsGoing + (10@10)].
            universe worlds at: tempWorld put: whereItsGoing.
            gathered add: tempWorld.
            self addConnection: tempWorld name]].
    tempPosition := aValue + (1@0).
    (tempPosition x < (1+GridSize x)) ifTrue: [(self isAnythingAt: tempPosition) ifFalse:
        [| tempWorld |
            tempWorld := GridWorld new.
            tempWorld universe: self universe.
            tempWorld name: (self getName).
            self pieces keysAndValuesDo: [:key :value | tempWorld pieces at: key put: value.

```

```

        key addDependent: tempWorld].
tempWorld addDependent: universe.
tempWorld movePieceFrom: aValue to: tempPosition.
worldPosition := (universe worlds at: self).
whereItsGoing := (worldPosition + (50@(index*50))).
[universe worlds values includes: whereItsGoing]
    whileTrue: [whereItsGoing:=whereItsGoing + (10@10)].
universe worlds at: tempWorld put: whereItsGoing.
gathered add: tempWorld.
self addConnection: tempWorld name]].
tempPosition := aValue - (1@0).
(tempPosition x > 0) ifTrue: [(self isAnythingAt: tempPosition) ifFalse:
    [!tempWorld !
        tempWorld := GridWorld new.
        tempWorld universe: self universe.
        tempWorld name: (self getName).
        self pieces keysAndValuesDo: [:key :value | tempWorld pieces at: key put: value.
            key addDependent: tempWorld].
        tempWorld addDependent: universe.
        tempWorld movePieceFrom: aValue to: tempPosition.
        worldPosition := (universe worlds at: self).
        whereItsGoing := (worldPosition + (100@(index*50))).
        [universe worlds values includes: whereItsGoing]
            whileTrue: [whereItsGoing:=whereItsGoing + (10@10)].
        universe worlds at: tempWorld put: whereItsGoing.
        gathered add: tempWorld.
        self addConnection: tempWorld name]]].

self changed: #worlds.
^gathered].!
```

```

generateWorlds: howManyTimes
    | newCreations |
```

```

Cursor wait show.
newCreations := OrderedCollection new.
newCreations add: self.
```

```

(howManyTimes = 1) ifTrue: [self generateWorlds]
    ifFalse: [
        (howManyTimes) timesRepeat: [ | gathering |
            gathering := OrderedCollection new.
            newCreations do: [:each | | temp |
                temp := OrderedCollection new.
                temp := each generateWorlds.
                temp do: [: aWorld | gathering add: aWorld]].
            newCreations := gathering]].
Cursor normal show.!
```

```

getName
    | n tempName |
    n:= 1.

    tempName := 'World 1'.

    [self nameChecksOut: tempName] whileTrue: [n:=n+1.
        tempName := 'World ', n displayString].

    ^tempName.!
```

```

gridSize
```

```

^GridSize!

gridSize: aPoint
  GridSize:= aPoint!

isAnythingAt: aPosition
  "Is this new position actually on the grid?"

  (aPosition > (0@0)) & (aPosition <= GridSize)
    ifFalse: [Dialog warn: 'This piece will fall off of the edge of the board.'.
      ^true].

  "Is there anything at this new position?"

  ^((pieces values) includes: aPosition).!

movePieceDown: aPiece
  | aPosition |

  aPosition := (pieces at: aPiece).
  self movePieceFrom: aPosition to: aPosition - (0@1).!

movePieceFrom: aPosition to: aNewPosition
  (self isAnythingAt: aNewPosition)
    ifFalse: [pieces keysAndValuesDo: [:key :value | value= aPosition
      ifTrue: [pieces at: key put: aNewPosition]].
      self changed: #pieces].!

movePieceLeft: aPiece
  | aPosition |

  aPosition := (pieces at: aPiece).
  self movePieceFrom: aPosition to: aPosition - (1@0).!

movePieceRight: aPiece
  | aPosition |

  aPosition := (pieces at: aPiece).
  self movePieceFrom: aPosition to: aPosition + (1@0).!

movePieceUp: aPiece
  | aPosition |

  aPosition := (pieces at: aPiece).
  self movePieceFrom: aPosition to: aPosition + (0@1).!

name
  ^name!

name: aName
  | oldName tempArray |
  oldName := (self name).
  name := aName.
  tempArray := Array new: 2.
  tempArray at: 1 put: oldName; at: 2 put: (self name).
  self changed: #name with: tempArray.!

nameChecksOut: aName
  | value |

  value := false.

```

```

universe worlds keys do: [:aWorld |
    (aWorld name = aName) ifTrue: {value := true}].
^value.!

pieces
    ^pieces!

pieces: aSet
    pieces:= aSet!

pieceToPlace
    ^pieceToPlace!

pieceToPlace: aPiece
    pieceToPlace := aPiece.!

position: aPiece
    (self pieces) keysDo: [: key | (key = aPiece name) ifTrue: {^pieces at: (aPiece name)}}!

removeConnection: aConnection
    | tempCollection |

    tempCollection := OrderedCollection new.
    connections do: [: each |
        each = aConnection ifFalse: {tempCollection add: each}].
    self connections: tempCollection.!

square
    ^SquareImage!

square: anImage
    SquareImage:= CachedImage on: anImage.!

tetrahedron
    ^TetrahedronImage!

tetrahedron: anImage
    TetrahedronImage:= CachedImage on: anImage.!

universe
    ^universe.!

universe: aUniverse
    universe:= aUniverse.!

without: anObject
    | tempDict aStream aPosition |

    tempDict := Dictionary new.
    aStream := ReadStream on: (anObject printString).
    aStream := aStream upToEnd.

    aPosition := (pieces at: anObject).
    (aPosition isNil) ifTrue: {^Dialog warn: 'The object ', aStream printString, ' is not present.'}.

    pieces keysAndValuesDo: [:key :value | (key = anObject) ifFalse:
        {tempDict at: key put: value}].

    pieces:=tempDict.
    self changed: #pieces.! !

!GridWorld methodsFor: 'initialize-release'!

```

initialize

```
"creates an empty dictionary and defines the boundaries of the gridWorld"
(GridSize isNil) ifTrue: [GridSize:= Point x: 6 y: 6].
pieces:= Dictionary new.
name := 'unnamed'.
connections:= OrderedCollection new.
```

```
self tetrahedron: (GridWorld tetrahedronImage).
```

```
self circle: (GridWorld circleImage).
```

```
self square: (GridWorld squareImage).
```

```
self pieceToPlace: 'empty'.
hasSpawned := false.!
```

!GridWorld methodsFor: 'testing'!

evaluate: aClaim

```
1 type tempClaim thisValue piece1 piece2 piece3 1
```

```
type := aClaim at: 1.
```

```
(type = $ ) ifTrue: [pieces keys do: [ : each 1
    (each name = (universe piece1)) ifTrue: [piece1 := each].
    (each name = (universe piece2)) ifTrue: [piece2 := each].
    (each name = (universe piece3)) ifTrue: [piece3 := each]].
```

```
piece1 isNil ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
    [GridWorld HasWarned: true.
```

```
    ^Dialog warn: 'An error has occurred in ', self name, '.\No piece called "" withCRs,
(universe piece1), "" was found.']].
```

```
piece2 isNil ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
    [GridWorld HasWarned: true.
```

```
    ^Dialog warn: 'An error has occurred in ', self name, '.\No piece called "" withCRs,
(universe piece2), "" was found.']].
```

```
tempClaim := String new: (aClaim size -1).
```

```
2 to: aClaim size do: [ :index1 tempClaim at: (index-1) put: (aClaim at: index)].
```

```
((piece1 isNil) 1 (piece2 isNil)) ifFalse:
```

```
[[((tempClaim at: 1) = $A) ifTrue: [thisValue := (self isThis: piece1 above: piece2)].
```

```
(((tempClaim at: 1) = $B) & ((tempClaim at: 3) = $I)) ifTrue:
```

```
    [thisValue := (self isThis: piece1 below: piece2)].
```

```
((tempClaim at: 1) = $L) ifTrue: [thisValue := (self isThis: piece1 leftOf: piece2)].
```

```
((tempClaim at: 1) = $R) ifTrue: [thisValue := (self isThis: piece1 rightOf: piece2)].
```

```
(((tempClaim at: 1) = $B) & ((tempClaim at: 3) = $t)) ifTrue:
```

```
    [piece3 isNil ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
```

```
    [GridWorld HasWarned: true.
```

```
    ^Dialog warn: 'An error has occurred in ', self name, '.\No piece called "" withCRs,
(universe piece3), "" was found.']]
```

```
    ifFalse: [thisValue := (self isThis: piece1 between: piece2 and: piece3)]]].
```

```
type = $C ifTrue: [ 1 checkSet 1
```

```
    checkSet := self getConnections.
```

```
    thisValue := false.
```

```
(System = #S4) ifTrue: [ 1 temp myConnections 1
```

```
    myConnections := Set new.
```

```
    universe worlds keys do: [:each1 (connections includes: (each name))
```

```
        ifTrue: [myConnections add: each]].
```

```

temp := Set new.
checkSet do: [: aConnection | (connections includes: (aConnection name))
    ifFalse: [temp add: aConnection]].
temp do: [: aWorld | | aReturn |
    aReturn := (aWorld evaluatePartTwo: tempClaim).
    (aReturn isNil) ifFalse: [aReturn ifTrue:
        [thisValue := true. (Example isNil) ifTrue: [Example:= aWorld]]]].
checkSet := myConnections].

checkSet isEmpty ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
    [GridWorld HasWarned: true.
    ^Dialog warn: 'You have not extended your diagram\far enough to interpret this
claim.\(' withCRs, self name displayString, ' was not connected to anything')]].
checkSet do: [: aWorld | | aReturn |
    aReturn := (aWorld evaluate: tempClaim).
    (aReturn isNil) ifTrue: [thisValue := nil]
    ifFalse: [aReturn ifTrue:
        [thisValue := true. (Example isNil) ifTrue: [Example:= aWorld]]]].

type = $M ifTrue: [ | checkSet |
    checkSet := self getConnections.
    thisValue := true.

(System = #S4) ifTrue: [| temp myConnections |
    myConnections := Set new.
    universe worlds keys do: [:each | (connections includes: (each name))
        ifTrue: [myConnections add: each]].
    temp := Set new.
    checkSet do: [: aConnection | (connections includes: (aConnection name))
        ifFalse: [temp add: aConnection]].
    temp do: [: aWorld | | aReturn |
        aReturn := (aWorld evaluatePartTwo: tempClaim).
        (aReturn isNil) ifFalse: [aReturn ifFalse:
            [thisValue := false. (Example isNil) ifTrue: [Example:= aWorld]]]].
    checkSet := myConnections].

checkSet isEmpty ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
    [GridWorld HasWarned: true.
    ^Dialog warn: 'You have not extended your diagram\far enough to interpret this
claim.\(' withCRs, self name displayString, ' was not connected to anything')]].
checkSet do: [: aWorld | | aReturn |
    aReturn := (aWorld evaluate: tempClaim).
    (aReturn isNil) ifTrue: [thisValue := nil]
    ifFalse: [aReturn ifFalse:
        [ thisValue := false. (Example isNil) ifTrue: [Example := aWorld]]]].

^thisValue!

evaluatePartTwo: aClaim

| type tempClaim thisValue piece1 piece2 piece3 |

type := aClaim at: 1.
(type = $ ) ifTrue: [pieces keys do: [: each |
    (each name = (universe piece1)) ifTrue: [piece1 := each].
    (each name = (universe piece2)) ifTrue: [piece2 := each].
    (each name = (universe piece3)) ifTrue: [piece3 := each]].
piece1 isNil ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
    [GridWorld HasWarned: true.
    ^Dialog warn: 'An error has occurred in ', self name, '.\No piece called "" withCRs,
(universe piece1), "" was found.']].

```



```

piece2 isNil ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
    [GridWorld HasWarned: true.
    ^Dialog warn: 'An error has occurred in ', self name, '.\No piece called "" withCRs,
(universe piece2), "" was found.']].
tempClaim := String new: (aClaim size -1).

2 to: aClaim size do: [ :index | tempClaim at: (index-1) put: (aClaim at: index)].

((piece1 isNil) | (piece2 isNil)) ifFalse:
[(((tempClaim at: 1) = $A) ifTrue: [thisValue := (self isThis: piece1 above: piece2)].
(((tempClaim at: 1) = $B) & (((tempClaim at: 3) = $I)) ifTrue:
    [thisValue := (self isThis: piece1 below: piece2)].
((tempClaim at: 1) = $L) ifTrue: [thisValue := (self isThis: piece1 leftOf: piece2)].
((tempClaim at: 1) = $R) ifTrue: [thisValue := (self isThis: piece1 rightOf: piece2)].
(((tempClaim at: 1) = $B) & (((tempClaim at: 3) = $t)) ifTrue:
    [piece3 isNil ifTrue: [thisValue:=nil. (GridWorld HasWarned) ifFalse:
    [GridWorld HasWarned: true.
    ^Dialog warn: 'An error has occurred in ', self name, '.\No piece called "" withCRs,
(universe piece3), "" was found.']]
    ifFalse: [thisValue := (self isThis: piece1 between: piece2 and: piece3)]]].

type = $C ifTrue: [ | checkSet |
    checkSet := self getConnections.
    thisValue := false.
    checkSet isEmpty ifTrue: [thisValue:=nil].
    checkSet do: [ :aWorld | | aReturn |
        aReturn := (aWorld evaluatePartTwo: tempClaim).
        (aReturn isNil) ifTrue: [thisValue := nil]
        ifFalse: [aReturn ifTrue:
            [thisValue := true]]].

type = $M ifTrue: [ | checkSet |
    checkSet := self getConnections.
    thisValue := true.
    checkSet isEmpty ifTrue: [thisValue:=nil].
    checkSet do: [ :aWorld | | aReturn |
        aReturn := (aWorld evaluatePartTwo: tempClaim).
        (aReturn isNil) ifTrue: [thisValue := nil]
        ifFalse: [aReturn ifFalse:
            [ thisValue := false]]].

^thisValue!

getConnections

| connectionsSet alreadyGot more |

connectionsSet := Set new.
more := true.
alreadyGot := Set new.
alreadyGot add: self.
universe worlds keys do: [:each | (connections includes: (each name))
    ifTrue: [connectionsSet add: each]].

(System ~= #T) ifTrue: [
    [more] whileTrue: [ | newBits collectedNames toAdd |
        more := false.
        newBits := Set new.
        toAdd := Set new.
        collectedNames := Set new.
        connectionsSet do: [ :each |

```

```

        each connections do: [ : element | newBits add: element]].
connectionsSet do: [ :each |
    collectedNames add: (each name)].
newBits do: [ : eachName |
    (collectedNames includes: eachName)
        ifFalse: [more:=true. toAdd add: eachName]].
universe worlds keys do: [ : each |
    (toAdd includes: (each name)) ifTrue: [connectionsSet add: each]]].

(System = #S5) ifTrue: [ more := true.
    [more] whileTrue: [ | newBit collectedNames toAdd |
        more := false.
        newBit := Set new.
        collectedNames := Set new.
        toAdd := Set new.
        universe worlds keys do: [ : each |
            (connectionsSet includes: each) ifFalse: [newBit add: each]].
        newBit do: [ :each |
            each connections do: [ : aConnection |
                connectionsSet do: [ : aWorld |
                    (aWorld name = aConnection)
                        ifTrue: [more:=true. connectionsSet add: each]]].
            connectionsSet do: [ : aWorld | collectedNames add: (aWorld name)].
            newBit do: [ :each |
                each connections do: [ : aConnection |
                    (collectedNames includes: aConnection)
                        ifFalse: [more:=true. toAdd add: aConnection]]].
            universe worlds keys do: [ :each |
                (toAdd includes: (each name)) ifTrue: [connectionsSet add: each]]].
        ^connectionsSet.].

```

isThis: aPiece above: bPiece

"Tests to see whether one piece is right of another or not."

```

| firstPosition secondPosition |
firstPosition := pieces at: (aPiece).
secondPosition := pieces at: (bPiece).
firstPosition y > secondPosition y
    ifTrue: [^true]
    ifFalse: [^false].!

```

isThis: aPiece below: bPiece

"Tests to see whether one piece is right of another or not."

```

| firstPosition secondPosition |
firstPosition := pieces at: (aPiece).
secondPosition := pieces at: (bPiece).
firstPosition y < secondPosition y
    ifTrue: [^true]
    ifFalse: [^false].!

```

isThis: aPiece between: bPiece and: cPiece

"Tests to see whether one piece is right of another or not."

```

| firstPosition secondPosition thirdPosition isIt |
firstPosition := pieces at: (aPiece).
secondPosition := pieces at: (bPiece).
thirdPosition := pieces at: (cPiece).
isIt := false.

```

"Runs through all four ways of pieces being between each other. Note that this could be simplified if the isThis:leftOf: and so on were changed so that they returned Boolean values rather than Dialog boxes."

```
(firstPosition y > secondPosition y) & (firstPosition y < thirdPosition y) ifTrue: [islt:= true].  
(firstPosition y < secondPosition y) & (firstPosition y > thirdPosition y) ifTrue: [islt:= true].  
(firstPosition x > secondPosition x) & (firstPosition x < thirdPosition x) ifTrue: [islt:= true].  
(firstPosition x < secondPosition x) & (firstPosition x > thirdPosition x) ifTrue: [islt:= true].
```

"returns a value based on the outcome of the testing."

```
islt    ifTrue: [^true]  
        ifFalse: [^false].!
```

isThis: aPiece leftOf: bPiece

"Tests to see whether one piece is right of another or not."

```
| firstPosition secondPosition |  
firstPosition := pieces at: (aPiece).  
secondPosition := pieces at: (bPiece).  
firstPosition x < secondPosition x  
    ifTrue: [^true]  
    ifFalse: [^false].!
```

isThis: aPiece rightOf: bPiece

"Tests to see whether one piece is right of another or not."

```
| firstPosition secondPosition |  
firstPosition := pieces at: (aPiece).  
secondPosition := pieces at: (bPiece).  
firstPosition x > secondPosition x  
    ifTrue: [^true]  
    ifFalse: [^false].! !
```

!GridWorld methodsFor: 'printing'!

printOn: aStream

```
aStream nextPutAll: (self name)! !
```

!GridWorld methodsFor: 'updating'!

update: anAspect

```
anAspect == #shape ifTrue: [self changed: #pieces].! !
```

!GridWorld methodsFor: 'private'!

> aWorld

```
^(self name > aWorld name)!
```

closeWindows

```
(self dependents) do: [: each | (each class = GridWorldInterface) ifTrue:  
    [each closeRequest]]!
```

sort: aSet

```
| changes aCollection |
```

```
changes := true.
```

```
aCollection := OrderedCollection new.
```

```
aSet do: [:each | aCollection add: each].
```

```

[changes] whileTrue: [|temp|
  changes:= false.
  1 to: (aCollection size-1) do: [:index|
    (aCollection at: index) > (aCollection at: index+1)
    ifTrue: [temp := aCollection at: index.
      aCollection at: index put: (aCollection at: index+1).
      aCollection at: (index+1) put: temp.
      changes := true]].
  ^aCollection.! !
"-----"!

```

```

GridWorld class
  instanceVariableNames: ''!

```

```

!GridWorld class methodsFor: 'instance creation'!

```

```

new
  ^super new initialize! !

```

```

!GridWorld class methodsFor: 'accessing'!

```

```

circleImage
  (CircleImage isNil) ifTrue: [
    CircleImage := (Image extent: 32@32 depth: 1 palette: MappedPalette whiteBlack
      bits: #[2r00000000 2r00000011 2r11000000 2r00000000
        2r00000000 2r00011100 2r00011100 2r00000000
        2r00000000 2r01100000 2r00000011 2r00000000
        2r00000001 2r10000000 2r00000000 2r11000000
        2r00000010 2r00000000 2r00000000 2r00100000
        2r00000100 2r00000000 2r00000000 2r00010000
        2r00001000 2r00000000 2r00000000 2r00001000
        2r00001000 2r00000000 2r00000000 2r00000100
        2r00010000 2r00000000 2r00000000 2r00000100
        2r00100000 2r00000000 2r00000000 2r00000010
        2r00100000 2r00000000 2r00000000 2r00000010
        2r00100000 2r00000000 2r00000000 2r00000010
        2r01000000 2r00000000 2r00000000 2r00000001
        2r01000000 2r00000000 2r00000000 2r00000001
        2r01000000 2r00000000 2r00000000 2r00000001
        2r01000000 2r00000000 2r00000000 2r00000001
        2r01000000 2r00000000 2r00000000 2r00000001
        2r00100000 2r00000000 2r00000000 2r00000010
        2r00100000 2r00000000 2r00000000 2r00000010
        2r00100000 2r00000000 2r00000000 2r00000010
        2r00010000 2r00000000 2r00000000 2r00000100
        2r00011000 2r00000000 2r00000000 2r00000100
        2r00001000 2r00000000 2r00000000 2r00001000
        2r00000100 2r00000000 2r00000000 2r00010000
        2r00000010 2r00000000 2r00000000 2r00100000
        2r00000001 2r10000000 2r00000000 2r11000000
        2r00000000 2r01100000 2r00000011 2r00000000
        2r00000000 2r00011100 2r00011100 2r00000000
        2r00000000 2r00000011 2r11100000 2r00000000] pad: 32)].
  ^CircleImage.!

```

Example

^Example!

Example: aValue

Example:= aValue!

GridSize

^GridSize!

GridSize: aPoint

GridSize := aPoint.

self changed: #pieces.!

HasWarned

^HasWarned!

HasWarned: aValue

HasWarned := aValue!

OnlyCircles

OnlyCircles isNil ifTrue: [OnlyCircles := true].

^OnlyCircles!

OnlyCircles: aValue

OnlyCircles := aValue!

squareImage

SquareImage isNil ifTrue: [

SquareImage := (Image extent: 32@32 depth: 1 palette: MappedPalette whiteBlack

bits: #[2r00000000 2r00000000 2r00000000 2r00000000

2r00000000 2r11111111 2r11111110 2r00000000

2r00000001 2r00000000 2r00000001 2r00000000

2r00000010 2r00000000 2r00000000 2r10000000

2r00001000 2r00000000 2r00000000 2r01000000

2r00010000 2r00000000 2r00000000 2r00100000

2r00011111 2r11111111 2r11111111 2r11110000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00010000 2r00000000 2r00000000 2r00010000

2r00011111 2r11111111 2r11111111 2r11110000

2r00000000 2r00000000 2r00000000 2r00000000] pad: 32)].

^SquareImage.!

System: aValue

System := aValue.!

tetrahedronImage

TetrahedronImage isNil ifTrue: [

TetrahedronImage := (Image extent: 32@32 depth: 1 palette: MappedPalette whiteBlack

bits: #[2r00000000 2r00000000 2r11000000 2r00000000

2r00000000 2r00000001 2r10100000 2r00000000

2r00000000 2r00000011 2r10100000 2r00000000

2r00000000 2r00000010 2r10010000 2r00000000

2r00000000 2r00000100 2r10010000 2r00000000

2r00000000 2r00001100 2r10001000 2r00000000

2r00000000 2r00001000 2r10001100 2r00000000

2r00000000 2r00010000 2r10000100 2r00000000

2r00000000 2r00110000 2r10000010 2r00000000

2r00000000 2r00100000 2r10000011 2r00000000

2r00000000 2r01000000 2r10000001 2r00000000

2r00000000 2r11000000 2r10000001 2r10000000

2r00000000 2r10000000 2r10000000 2r10000000

2r00000001 2r00000000 2r10000000 2r01000000

2r00000011 2r00000000 2r10000000 2r01100000

2r00000010 2r00000000 2r10000000 2r00100000

2r00000100 2r00000000 2r10000000 2r00010000

2r00001100 2r00000000 2r10000000 2r00011000

2r00001000 2r00000000 2r10000000 2r00001000

2r00010000 2r00000000 2r10000000 2r00001000

2r00110000 2r00000000 2r10000000 2r00000100

2r00100000 2r00000000 2r10000000 2r00000110

2r01000000 2r00000000 2r10000000 2r00000010

2r11110000 2r00000000 2r10000000 2r00000001

2r00111100 2r00000000 2r10000000 2r00000011

2r00001111 2r00000000 2r10000000 2r00011110

2r00000011 2r11000000 2r10000000 2r01111000

2r00000000 2r11110000 2r10000001 2r11100000

2r00000000 2r00111100 2r10000111 2r10000000

2r00000000 2r00001111 2r10011110 2r00000000

2r00000000 2r00000011 2r11110000 2r00000000

2r00000000 2r00000000 2r11000000 2r00000000] pad: 32)].

^TetrahedronImage.!

```
Object subclass: #Piece
  instanceVariableNames: 'shape name '
  classVariableNames: ''
  poolDictionaries: ''
  category: 'MLTool'!
```

```
!Piece methodsFor: 'accessing'!
```

```
beCircle
  shape:= 'circle'.
  self changed: #shape!
```

```
beSquare
  shape:= 'cube'.
  self changed: #shape!
```

```
beTriangle
  shape:= 'tetrahedron'.
  self changed: #shape!
```

```
name
  ^name!
```

```
name: aName
  name:= aName.
  self changed: #shape!
```

```
shape
  ^shape!
```

```
shape: aShape
  shape:= aShape.
  self changed: #shape! !
```

```
!Piece methodsFor: 'initialize-release'!
```

```
initialize
  "Set up the piece with default name and shape"

  name:= 'unnamed'.
  self beSquare.! !
```

```
!Piece methodsFor: 'printing'!
```

```
printOn: aStream
  "Adapts the printOn: message to display the information about this piece"

  aStream nextPutAll: 'A ', shape displayString, ' called ', name displayString.! !
```

```
!Piece methodsFor: 'testing'!
```

```
isCircle
  ^(self shape = 'circle')!
```

```
isSquare
  ^(self shape = 'cube')!
```

```
isTriangle
  ^(self shape = 'tetrahedron')! !
```

!Piece methodsFor: 'private'!

> aPiece
 ^{^(self name > aPiece name)!}!
"-----"!

Piece class
 instanceVariableNames: ""!

!Piece class methodsFor: 'instance creation'!

new
 ^{^super new initialize!}!

Appendix 7: Observational logs, second study

This appendix contains the time-indexed observational logs from the five subjects who took part in the formative evaluation of MoLE, described in chapter five.

Subject 1

0:00 Introduce task
0:20 Opens first universe window
0:50 Clicks creation button, can't work out problem.
1:00 Places world, drags & drops, and opens it.
1:40 "Am I supposed to be trying to break this?"
1:50 Drags a piece off a grid, then fills grid up, trying to crash program
2:20 Tries dragging piece from one world to another, out of window.
2:40 Tries placing one piece on top of another.
3:20 Can't call an active window to the front - problem using X-Windows.
3:30 "Oh! I've got a connection"
3:50 Interrogates pieces
4:10 "Why does it let you put the same piece twice?"
5:27 "I can't get that one out" (referring to 3rd piece, used for 'between' queries, but nothing else.
6:22 "No piece called Piece 7 was found." (Misconception in use of Modal operators, that they apply to starting world, rather than diagram)
7:04 Subject doesn't understand response to an error: "It says World 2, but I'm in world 1."
7:26 Misconception resolved: "Ah! Of course!"
8:30 Repeats attempts to use Modal queries, successfully.
9:30 Confusion about X-Windows term 'close' (i.e. minimise).
9:50 Prompt subject to try and build diagrams and work through a problem.
10:20 Subject can't find the 'Move Pieces' command in the menus.
10:50 Problems using X-Windows scrolling.
11:50 Suggests way of resetting universe; these are actually included already.
12:40 "How do you get them off?" (deleting pieces)
13:43 Tries placing 21 circles on a 6x6 grid to break system
14:10 Confusion over delay as system generates diagram
14:20 Problems with X-windows scroller
16:15 Decides to try crashing software
17:00 Wonders why claims true or false depending on the world selected. Observer explains Modal systems.
18:40 Alters system, delighted to find "it's true!"
19:00 Observer talks through the answer for subject
19:49 Subject tries another problem
21:07 Changes qualities of piece in one world to see effect this has on others.
23:29 Confusion over why cursor position moves when a window is opened. (X-windows)
24:36 Tries to crash program by opening 32 windows. Program holds. Closes windows.
24:53 "Well, I'm very impressed with that. Very robust."
25:35 Resets universe. Discuss how software will be used by students.
28:00 Subject starts using software again.
28:40 Subject asks again about systems.
29:07 Sets up another diagram
29:40 Problem with double clicking.
31:00 Difference between 'Must' and 'can' claims
31:40 Extend diagram as a layer
31:50 Overlapping worlds leads to confusion: "They're the same, so they've got a shared link."
32:30 Talk subject through systems again.
34:45 Tries changing name of world.
35:35 Tries long name (35+ characters)
35:40 System breaks; problem with drawing links to world which no longer exists, as name has altered.

Subject 2

- 0:00 Subject chooses to work through problem
- 0:30 Clicks world creation button, nothing happens.
- 0:40 Places world, drags & drops, and opens it.
- 1:05 Finishes examining menu and buttons, tries to place a piece.
- 1:55 Several pieces placed. Unsure about how to proceed.
- 2:30 Places another world.
- 3:05 Finishes editing second world.
- 3:10 Tries to close world; problem with X-windows ('minimise').
- 3:25 Links one world to another.
- 3:45 Tries dragging worlds in universe window to see if the link updates.
- 4:10 Interrogates pieces. Error returned. (piece not found)
- 4:15 Inspect worlds.
- 4:40 Deletes second world.
- 4:45 Opens first world.
- 4:55 Uses "move pieces" command to generate a diagram.
- 5:20 Successfully enters a relational query.
- 5:55 Fails to enter a correct Modal query (diagram doesn't have enough layers)
- 6:20 Fails again.
- 6:25 Query about Modal operators; observer explains.
- 7:25 Repeats attempts to use Modal queries, successfully.
- 7:45 Starts working through worlds one at a time to extend diagram.
- 9:50 Tries longer Modal claim, successfully.
- 10:15 Subject suggests ordering buttons as: Set up, reset, Add a world, Delete a world. (currently add, delete, set up, reset) This to support order in which buttons are used, and to place commonly used buttons near sentence entry commands.
- 11:40 Looks at set-up dialogue; changes system
- 12:05 Tries another claim.
- 12:55 Tries a further claim.
- 14:10 Returns to dialogue window; changes system several times.
- 14:25 Tries changing grid size; warned that you cannot do this when worlds have been created.
- 14:40 Resets universe.
- 14:55 Changes grid size.
- 15:15 Places pieces in 4x3 grid.
- 16:10 Creates diagram.
- 16:25 Resets universe.
- 16:40 Alters grid size.
- 17:05 Places three worlds.
- 17:30 Session ends.

Subject 3

- 0:00 Subject chooses to work through an example.
- 1:25 Confusion as clicks on "add a world" button, but nothing happens.
- 1:40 Observer intervenes, explaining how to place worlds.
- 2:20 World placed.
- 2:30 World opened. Confusion when trying to place pieces.
- 3:10 Observer intervenes, explaining how to place pieces.
- 4:20 Pieces placed. Subject unsure how to continue. Observer suggests trying claims.
- 4:50 Subject tries Modal claim; fails. (World not entered)
- 5:15 Re-tries claim; world entered successfully, but claim fails (diagram not extended far enough)
- 5:50 Tries relational claim; succeeds.
- 6:10 Second world created and pieces placed.
- 7:25 Subject links worlds.
- 7:50 Subject tries more claims; relation and some one-operator Modal succeed.
- 9:30 Confusion trying to close world (X-windows; minimise).
- 10:20 Subject deletes both worlds, starts again.
- 10:55 Subject creates a new world, places pieces.
- 11:20 Subject uses "move pieces" command to build diagram.

11:50 Subject tries Modal claim; prompted by software to extend diagram.
 13:10 Diagram extended one layer.
 13:20 Attempts two Modal operators in a claim; succeeds.
 13:50 Observer asked to explain Modal logic.
 15:20 Subject compares worlds, to view changes. Opens five worlds.
 15:40 Confusion over close (X-windows)
 16:20 Subject queries why several worlds are the same (identity and inverse operations)
 17:40 Subject re-sets tool.
 18:05 End of session

Subject 4

0:00 Chooses to start by breaking software
 0:50 Confusion over additional X-windows menu
 1:30 "Ah; what's that?" (X-windows outline of new window for placing & re-sizing.
 1:50 X-windows confusion; close & minimise.
 2:45 Add a new world button causes confusion with no immediate effect apparent.
 4:10 Deleting a world doesn't close the associated window.
 4:20 Subject selects a delete dialogue box. "The thing about clicking buttons is that you always expect something to happen."
 5:30 Reset universe doesn't close associated world windows.
 5:45 Confusion with close & minimise (X-Windows)
 6:25 Confusion over re-sizing windows (X-windows)
 6:40 Subject starts placing pieces in worlds
 6:50 (Of universe representation) "Shame that doesn't indicate it's got pieces in as well. I guess it's the size issue. It would be cluttered, wouldn't it."
 8:40 Tries to build a sentence.
 8:50 Confusion over order of building sentences.
 9:55 Successfully builds a sentence
 10:30 "Do I need to put more in my worlds?"
 10:45 Observer explains the layers of the diagram to the user.
 11:10 Tries to add new worlds by creating an empty one & copying pieces' positions. Observer explains use of 'generate worlds' command.
 12:10 "Why are there two new worlds?"
 12:40 "How do you maximise this?" (X-windows)
 13:00 "Why are these two worlds the same?" (identity relation)
 13:50 Tries evaluating a new claim.
 14:30 "True! See?"
 14:40 Tries a further claim.
 15:10 Comments on use of menus to enter claims being nice for users.
 15:30 Offered an example by MoLE to explain why the claim was false; refuses it.
 16:00 Tries dropping pieces off of the edge of the board.
 16:30 Tries tapping keys & buttons at random to break software.
 17:20 "This is dinky!"
 17:30 System crash after attempt to change a world's name.
 18:00 System reset; subject continues trying to break software.
 18:25 "It's very dinky. I really liked it. It's so tidy, so nice when you see properly designed things."
 19:30 "It is genuinely hard to make an error. It is quite tough! It's not slow, is it?"
 19:45 Session ends.

Subject 5

0:00 Opts to break the software
 0:45 Confusion over X-windows menu
 1:20 Clicks on "add a world" button; confused when nothing happens.
 2:10 Tries to use set-up dialogue.
 2:20 Instruct subject how to place & open worlds.
 2:50 Manages to open a world
 3:30 Places pieces on grid
 3:35 Connects world to itself.
 3:45 Confusion over which window is active (X-windows)
 4:05 Tries hitting buttons across the keyboard to see what happens; software remains intact.

4:20 Places more worlds
 4:30 "How many of these can I do?"
 4:50 Gives up after 17 worlds.
 5:00 Tries to type in the sentence window. Hasn't understood use of buttons.
 5:30 Places pieces in a new world.
 6:05 Tries entering claims
 6:25 "It's not obvious where you should put your sentences" (hasn't yet chosen a world, preventing use of buttons, and fails to understand warning prompts)
 6:47 Selects world, builds a sentence successfully.
 7:50 Gets sentence value back.
 8:20 Resets the tool; starts again.
 9:00 Builds new world and generates diagrams
 9:40 Tries entering a sentence
 10:00 Encounters problem with order of entering sentence.
 10:35 View the example offered as an explanation of the result
 11:20 Program warns about the claim being too long for the diagram that's been built.
 11:50 Views another example from another successful claim.
 12:30 Opens worlds and extends diagram
 13:00 Deletes worlds
 13:15 Attempts to use stacked operators
 13:50 Close & minimise confusion
 14:15 Views movement by opening & comparing worlds
 14:50 Opens four worlds
 15:00 Observer explains identity condition
 15:25 Close/minimise confusion, twice.
 15:50 Fills in worlds and attempts to extend it.
 17:15 Finished placing pieces on the grid
 17:38 Tries out claims on new diagram
 18:10 Stopped by not having enough layers to cope with that many Modal claims
 19:40 Observer explains systems
 20:40 Tries to break program by deleting intermediate worlds to see if links fail.
 22:00 End of session.

Appendix 8: Study material, third study

This appendix contains the pre-test, instructional material, exercise sheet, software prompt sheet, and post-test, as used in the study.

Pre-Test: Part A

Name: Date:.....

Male..... ☐ Female: ☐

Have you studied any logic before?

No, not at all..... ☐

I'm studying a course at the moment..... ☐

I've completed a course in logic..... ☐

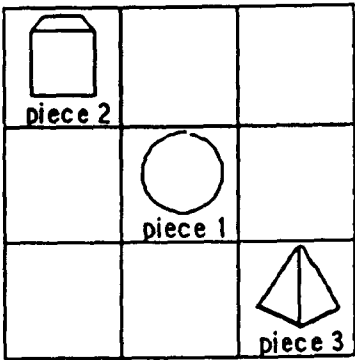
I've completed a course in logic and studied a more advance type
of logic, too (e.g. modal logic, fuzzy logic, etc)..... ☐

Pre-Test: Please attempt any questions you think you might be able to answer. Use extra sheets of paper if you need to, and hand them in with this test.

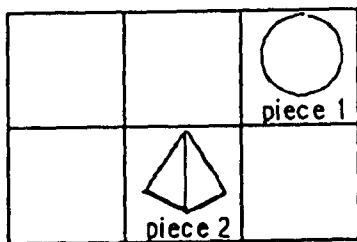
- 1) Give a brief explanation of what you think "accessibility relations" are.

- 2) Briefly explain the differences between the Modal systems T, S4 and S5.

- 3) Given the following diagram, prove (in system T) that it must be the case that piece 1 is between piece 2 and piece 3. Piece 1 is allowed to move up to one square left, right, up or down. "piece 1 is between piece 2 and piece 3" should be understood to mean that piece 1 is in a column which is to the right of one piece and to the left of the other, or else that it is in a row which is above one piece and below the other.

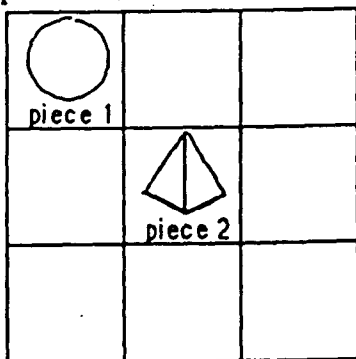


- 4) Given the following diagram, show that the claim, "It can be the case that piece 1 is left of piece 2" can be true in S4 and S5, but not in T, given the accessibility relation that piece 1 can move up, down, left or right by up to one square in any one movement. (Hint: the accessibility relation can be applied more than once)



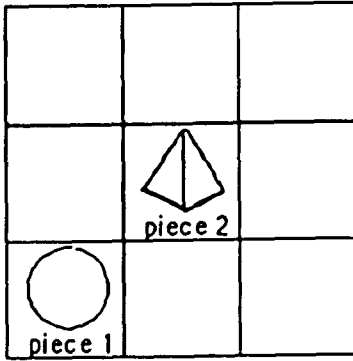
- 5) For each of the following claims, state in which Modal systems (T, S4, S5) they will be valid. (N.B. M is shorthand for, "it must be the case that...", C is shorthand for, "it can be the case that...", and the symbol " \rightarrow " means "implies")

- a) MC "piece 1 is above piece 2" is true if MMCMC "piece 1 is above piece 2", for the situation:



(i.e. that MMCMC "piece 1 is above piece 2" \rightarrow MC "piece 1 is above piece 2")

- b) M "piece 1 is above piece 2" is true if CCM "piece 1 is above piece 2", for the situation:



(i.e. that CCM "piece 1 is above piece 2" \rightarrow M "piece 1 is above piece 2")

- c) CMp is true if CMMMCMp is true, where p is shorthand for any true claim.

(i.e. CMMMCMp \rightarrow CMp)

- d) CMp is true if CCCp is true, where p is shorthand for any true claim.

(i.e. CCCp \rightarrow CMp)

- 6) Let U be a new modal system, in which situations always change. In other words, when evaluating claims, you are not allowed to consider whether or not the claim is true in your starting world. Put another way, the system is not reflexive.

Give an example of when such a system could occur. Show that, for a claim p which is true in world 1 (your starting world), the claim "It can be the case that p" can be false. (In other words, show that $p \rightarrow Cp$ is no longer true)

What is Modal Logic?

Modal logic is an extension of first-order logic. First-order logic deals with two concepts: the validity of a conclusion, and the truth of sentences. Whilst related, these are distinct. If you assume that the sky is covered in material, and that all material is paisley, it's logically valid to conclude that the sky is paisley, even though your assumptions are wrong.

However, sentences such as "The sky is blue" can change their truth value (it might be cloudy and grey, for example). First-Order logic cannot cope with changes; it merely describes what is actually the case for some given situation. Nor can it cope with alternative situations (such as, "What if...?"). The sky may be blue today, but did it have to turn out this way?

These types of problems are called **contingencies**. Contingent sentences can be either true or false, depending on the situations they are evaluated in. For a sentence to be contingent, we need to come up with at least one possible situation in which it can be true, and one in which it can be false. The statement, "All redcurrants are red" is contingent, because we can image a world in which all redcurrants *are* red (because they've all ripened), and one where some of them are still green and unripe.

Because first order logic is not designed to deal with contingencies, Modal logic was created. This introduces **Possible Worlds**, which allow you to describe alternative situations as a set of statements. Each different situation is a new possible world. To give some examples, one possible world would be a place identical to this world except that the sky is grey; another would be identical except for having a red sky; another would be identical except that it is snowing; and so on.

What is it that possible worlds allow us to do? Well, contingent sentences can be either true or false, depending on the situation. Possible worlds provide a way of distinguishing contingent claims from non-contingent ones. Modal logic does this by introducing two **Modal operators**, **Necessity** and **Possibility**. These are quite straightforward: something is necessary if it must happen, and possible if it can happen. In terms of possible worlds, this gives us the following definitions:

- A claim is **necessarily true** (must happen) if it is true in all accessible possible worlds.
- A claim is **necessarily false** (cannot happen) if it is false in all accessible possible worlds.
- A claim is **possibly true** (can happen) if it is contingent; in other words, it is true in at least one accessible possible world, and false in another one.

Since it can be cumbersome to write, "it is necessarily true that..." or "it is possibly true that..." before claims, these are abbreviated to M (for, "it must be the case that...") and C (for, "it can be the case that...").

How do we decide what counts as an 'accessible' possible world and what doesn't? Is it permissible to include a world where the sky's green with pink polka dots? If not, why not? These decisions are taken on by the **Accessibility Relation**. This is some function which transforms the situation you start with into new possible worlds. Depending on what your possible world is, almost anything can be an accessibility relation. "Add one" is a perfectly good relation if you're talking about numerical values; "make the sky look different" is fine for the examples mentioned above. All that is required is that the relation can be applied to one situation in order to decide what other worlds are possible.

One problem with possible worlds is that there might be an infinite number of them, since almost anything can be altered. In order to make life simpler, a tightly constrained world will be used for the examples in this material. Specifically, we are going to talk about grids with pieces on them, which we will call "grid worlds".

Grid worlds look like this:

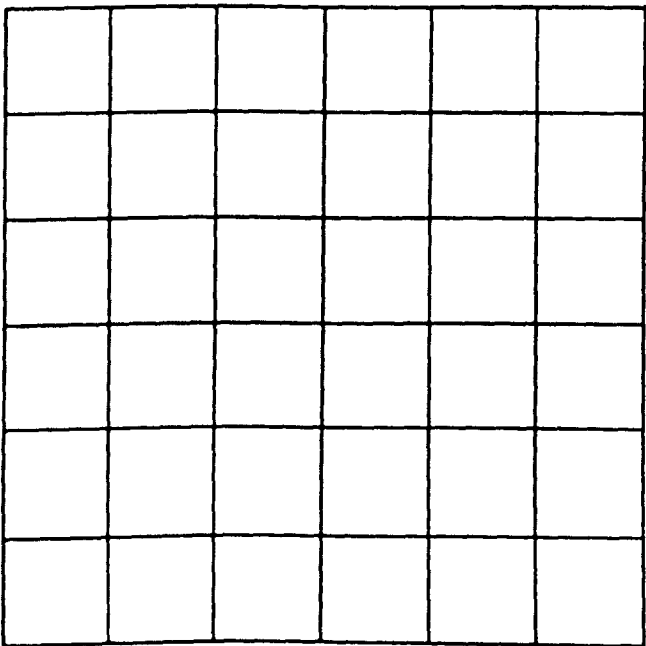


Figure 1: the grid from a "grid world"

Grids can have different sizes, but once we've specified what size each one is, it won't be allowed to change shape. So, a 2x3 grid world will remain 2 squares long and 3 squares high at all times.

On each grid world we can have three types of object: spheres, cubes, and tetrahedrons. A grid world with one sphere and two tetrahedrons on it might look like this:

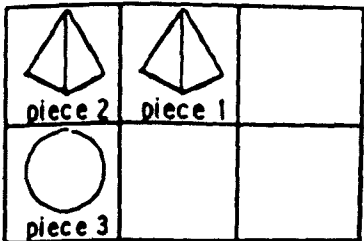


Figure 2: A grid world with one sphere and two tetrahedrons

Now try Exercise 1.

Now that our possible worlds have been defined, we need to specify an accessibility relation which can be used to create new situations. Spheres are allowed to roll around the grid; other pieces are not. At each step, any one sphere can move one square up, down, left or right, so long as it doesn't roll off the edge of the board or into a square that is already occupied. We also include the identity relation, "do nothing", as part of our accessibility — this covers situations when, although pieces have had the chance to move, none of them actually does.

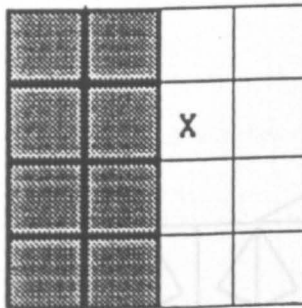
When a sphere is allowed to move, several new worlds are created, one for each position that the piece could end up in. To show that these new worlds have been created by applying our accessibility relation a line is drawn from our starting situation to each of the new grid worlds which has been created. A series of grid worlds linked by lines in this way is called a **Modal diagram**.

Now do Exercise 2

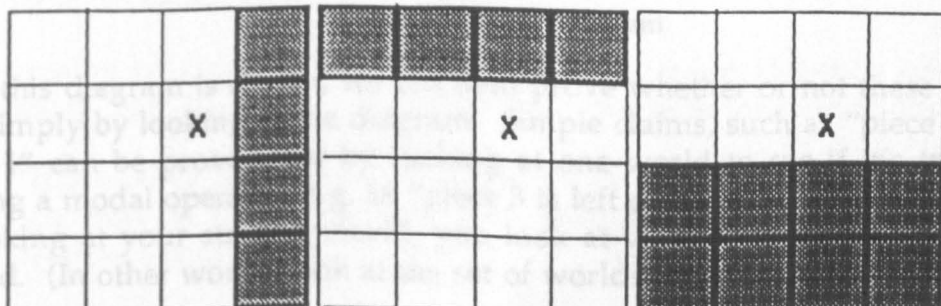
Modal Proofs

To avoid ambiguity, we will only allow certain simple questions to be asked about this world. They are: is one piece left of, right of, above, or below another piece? Also, by combining these, we can ask if a piece is between two other pieces. It's important to know precisely what these things mean, in order to avoid misinterpretation. Given below are definitions and examples.

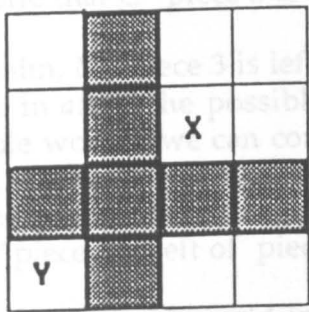
- "The circle is left of X" — the circle is in a row left of the one with X in it.



- "Right of", "Above", "Below" — these are all defined in a similar way to "Left of", so that they won't be true if the circle's just in line with a letter. This means that it's impossible for a circle to be left of *and* right of the same object.



- "The circle is between A and B" — this will only be true if the circle is right of A and left of B (assuming A is to the left of B), or if it is below A and above B (assuming that A is above B).



We have now covered enough material to allow you to construct Modal proofs. A proof is a diagram which shows whether or not a claim is true. If you look back at figure 2, you'll see that this is a proof that, given this world as a starting condition, piece 3 is left of piece 1.

But does piece 3 *have* to be left of piece 1? Can piece 3 ever be right of piece 1? Questions such as these introduce the modal operators necessity and possibility. In effect, they are asking whether the claims, M "piece 3 is left of piece 1" and C "piece 3 is right of piece 1" are true.

To assess the validity of these claims, we need to construct a modal diagram, starting with the initial condition given in 2, and allowing pieces to move. This is shown in figure 3.

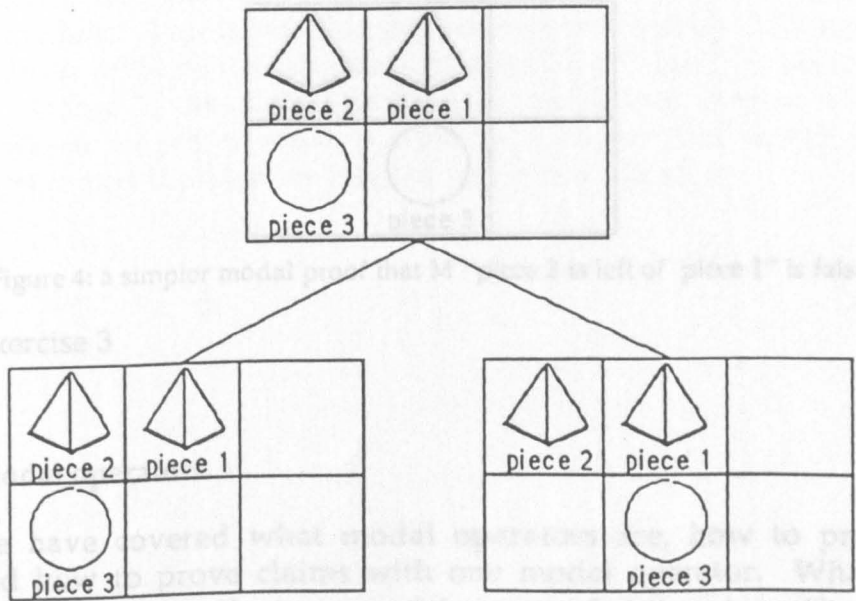


Figure 3: a Modal diagram

Once this diagram is drawn, we can then prove whether or not these claims are true simply by looking at the diagram. Simple claims, such as "piece 3 is left of piece 1" can be proved just by looking at one world to see if it's true or not. Adding a modal operator (e.g. M "piece 3 is left of piece 1") means that instead of looking at your starting world, you look at worlds in the next 'step' down instead. (In other words, look at the set of worlds created by moving the pieces)

The claim, C “piece 3 is left of piece 1”, will be true if “piece 3 is left of piece 1” is true in *at least one* of the possible worlds in the next ‘step’. (If we found that “piece 3 is left of piece 1” wasn’t true in any of the possible worlds, we can conclude that C “piece 3 is left of piece 1” is an invalid claim)

The claim, M “piece 3 is left of piece 1”, will be true if “piece 3 is left of piece 1” is true in *all* of the possible worlds in the next ‘step’. If it isn’t true in all the possible worlds, we can conclude that M “piece 3 is left of piece 1” is an invalid claim. Because of this, as soon as you find a counter-example (a world in which “piece 3 is left of piece 1” is false), you can stop drawing the diagram — M “piece 3 is left of piece 1” will be invalid because of this world.

So, given the grid world in figure 2, we have just proved that C “piece 3 is left of piece 1” is valid, and M “piece 3 is left of piece 1” is invalid. In fact, to show that M “piece 3 is left of piece 1” isn’t valid, we only needed to have drawn this:

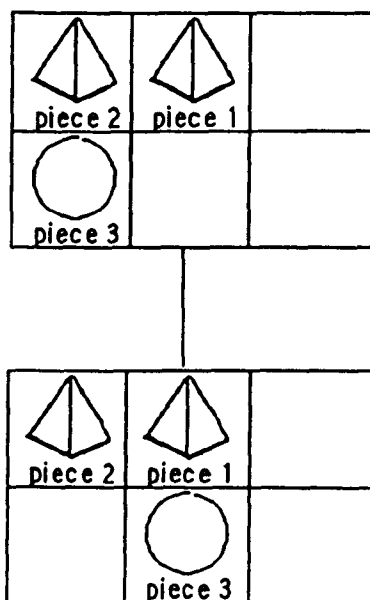


Figure 4: a simpler modal proof that M “piece 3 is left of piece 1” is false.

Now do Exercise 3

Stacked Modal operators

So far, we have covered what modal operators are, how to prove simple claims, and how to prove claims with one modal operator. What happens, though, if we have more than one modal operator? What does, “it must be the case that it can be the case that...” mean?

The meaning of claims with more than one operator depends on what you understand “can” to mean. There are three common interpretations:

- “It can be the case that...” refers only to what is possible at the next ‘step’. (i.e. only in the worlds you can reach after applying the accessibility relation once)

- “It can be the case that...” means, “at some point in the future, this claim will be true” (i.e. it happens in one of the worlds you can reach by applying the modal operator more than once)
- “It can be the case that...” means, “can it ever be the case that...”; in other words, can this claim be true in any of the worlds we can reach in the future, *or* in any of the worlds which could have happened before this point? In other words, we ask all three of these questions: Was this claim ever true? Is it true? Will it ever be true in the future?

Formally, these three different views are treated as different systems, called T, S4 and S5. These will be introduced one at a time. All three use the same type of diagrams to prove claims; however, these diagrams are “read” in slightly different ways for each of the systems.

System T: Will it be true at the next step?

In system T, each claim only looks at the next ‘step’ of the diagram. It’s as if you can see what’s coming up next, but can’t predict what might happen after that.

To recap, “piece 3 is left of piece 1” only needs you to look at the world you start in, and C “piece 3 is left of piece 1” needs you to see if “piece 3 is left of piece 1” is true in any of the worlds you can get to after applying the accessibility relation once. Adding more modal operators extends this process.

In order to see whether the claim CC “piece 3 is left of piece 1” is true, we need to extend the diagram shown in figure 3. This gives us a new diagram (figure 5). Note that even when two worlds look the same, they’re treated as being different situations. This is because they’ve been created by different routes. It can also help to think of the diagrams as starting at time 0, with the first step happening at time 1, the second at time 2, and so on; because of this, even worlds in which no pieces move are different to the grid world you started with (they have had the identity relation applied to them).

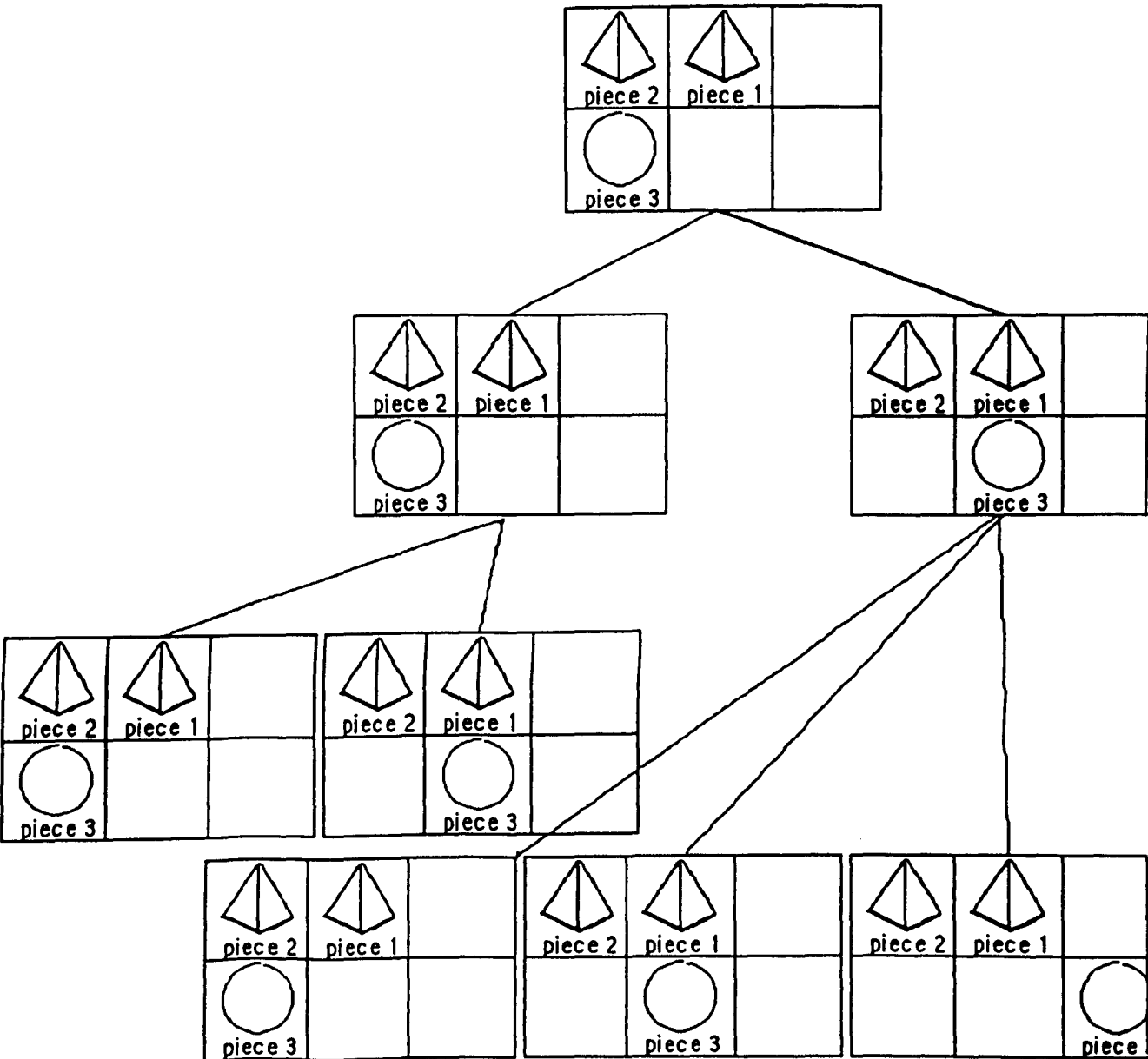


Figure 5: an extended modal diagram

We can then evaluate CC “piece 3 is left of piece 1” in the top world. This involves seeing if C “piece 3 is left of piece 1” is true in any of the worlds in the first ‘step’ down. In each of the worlds in this step, we then see if “piece 3 is left of piece 1” is true for one of the worlds in the second step down.

In other words, each modal operator you add means you need to add another step to the diagram. This process is shown in figure 6.

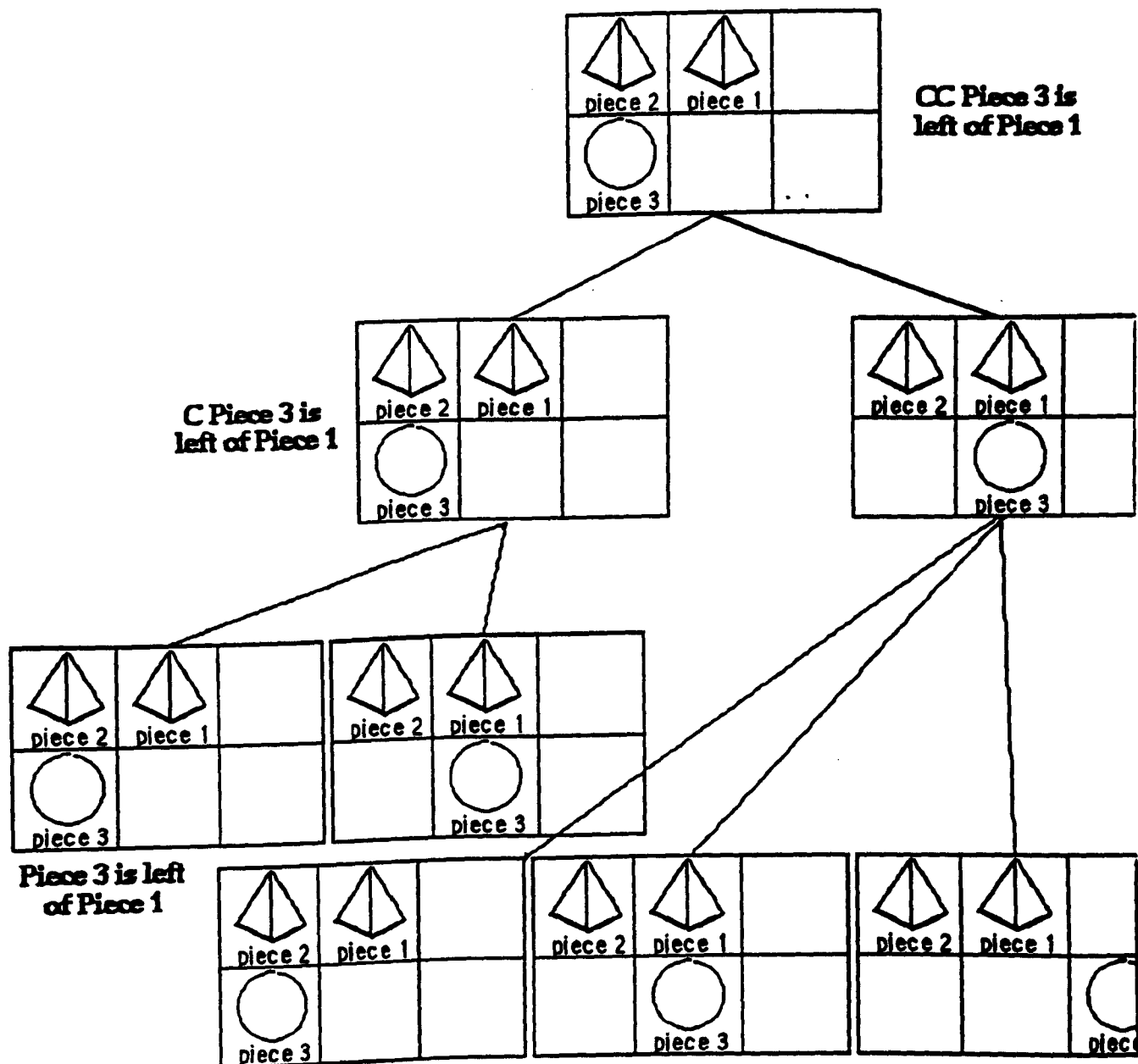


Figure 6: a modal diagram, with the claims which need to be applied marked on next to each level

Now do Exercise 4

System S4: Will it be true in the future?

System S4 asks a slightly different type of question from T. It's more like asking, "can we get there from here?", rather than "can it happen next?" In S4, you don't just look at the next step — you look at *all* the steps ahead, as far as your diagram goes. Because of this, it's the hardest of the three systems to apply.

In system T, we learnt how to deal with claims like, CC "piece 3 is left of piece 1". Each layer of the diagram was given one claim to deal with. At the first layer, this was C "piece 3 is left of piece 1", and at the second, "piece 3 is left of piece 1".

In S4, because we have to look at more than just the next step, something interesting happens. To see if CC “piece 3 is left of piece 1” is true, we need to check that C “piece 3 is left of piece 1” is true in *all* the possible worlds which happen below this level. This means not only the 2 worlds at the second step of the diagram, but also the 5 at the third step, and so on, as far as the diagram extends.

This is where it gets complicated. To see if C “piece 3 is left of piece 1” is true in each world, we need to see if “piece 3 is left of piece 1” is true in all the worlds ahead of that. In other words, for each world at the second step, we need to see if “piece 3 is left of piece 1” is true for worlds in step three and beyond. For all the worlds in step three, we need to see if “piece 3 is left of piece 1” is true in step four and beyond. And so on.

Sometimes, it won't make sense to see if a claim is valid. For example, if a grid world isn't connected to anything else, you can't ask it if C “piece 3 is left of piece 1” is true. What you need to do in cases like this is ignore them — if applying a claim doesn't make sense, you just don't do it.

What effect does all this looking into the future have? Well, it means that the logic can spot patterns which repeat themselves. For example, in the diagram we've been using (see figure 6), certain patterns of worlds crop up over and over again.

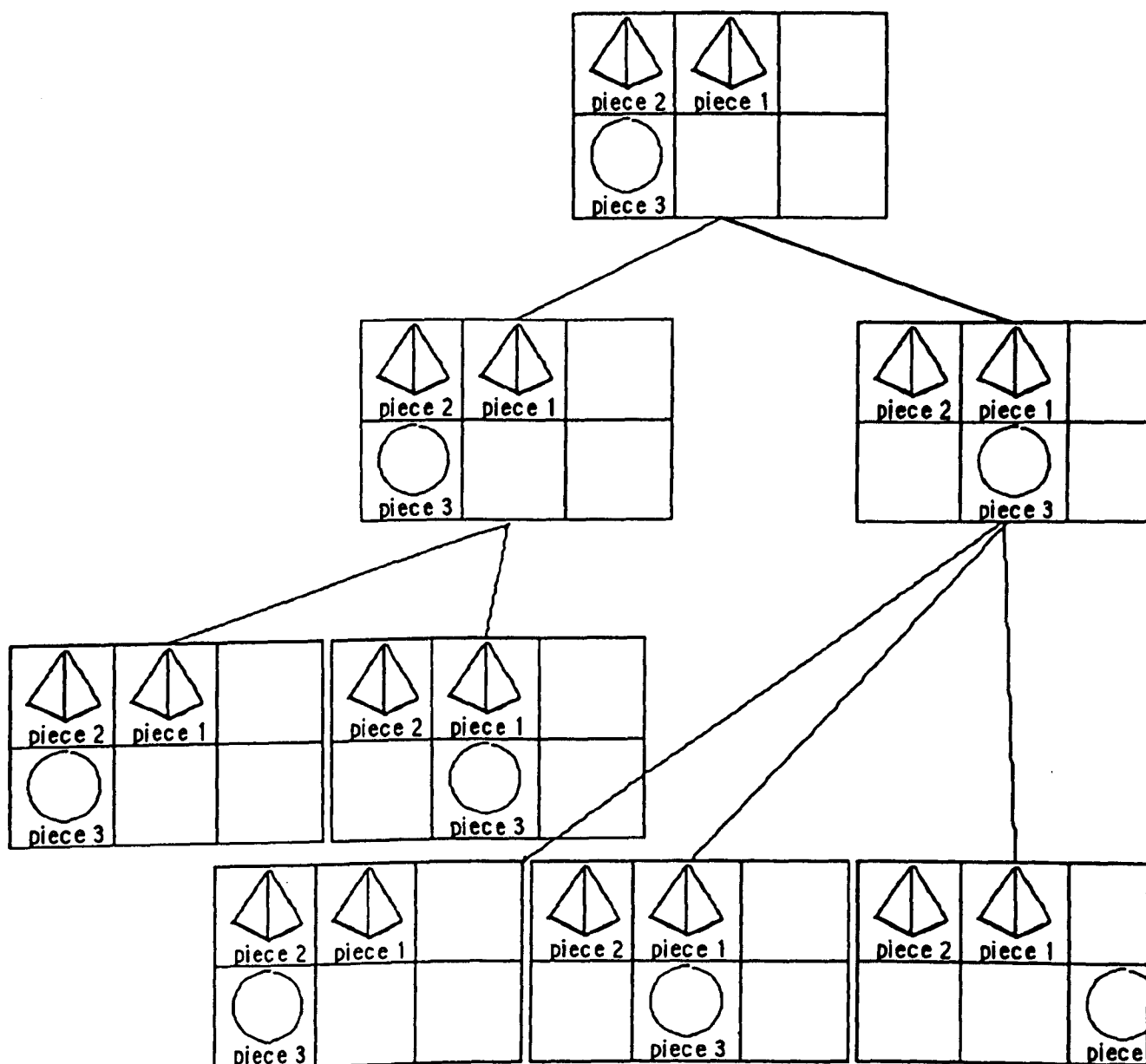


Figure 6: the extended modal diagram

If we extended the diagram even further, you can image how these patterns would keep repeating themselves. This means that the truth values of claims made about those worlds would keep repeating themselves, too. Effectively, we could just keep adding that pattern again and again, and the claim would still be true. So, if MC "piece 3 is left of piece 1" is true, so will be MCMC "piece 3 is left of piece 1", and MCMCMC "piece 3 is left of piece 1", and so on. It's as if we've put a loop into the diagram, which we can just carry on going around for as long as we want, simply adding the same modal operators each time.

Also, for MCMC "piece 3 is left of piece 1" to be true, MC "piece 3 is left of piece 1" will also be true. This is because finding out that MCMC "piece 3 is left of piece 1" is true has already involved checking that MC "piece 3 is left of piece 1" in the future.

Now, something useful happens. We know that:

- MCMC “piece 3 is left of piece 1” implies that MC “piece 3 is left of piece 1” is true, and
- MC “piece 3 is left of piece 1” also implies that MCMC “piece 3 is left of piece 1” is true

Since each of these claims implies the other, we can swap them over whenever we want to. The same is true for any pattern of modal operators — for example, we can swap M “piece 3 is left of piece 1” for MM “piece 3 is left of piece 1”, or C “piece 3 is left of piece 1” for CC “piece 3 is left of piece 1”, or MCMCMC “piece 3 is left of piece 1” for MCM “piece 3 is left of piece 1”.

Being able to swap patterns in this way means that we can simplify stacks of modal operators. Any repeated pattern of operators can be cancelled out, so that MM “piece 3 is left of piece 1” will be shortened to M “piece 3 is left of piece 1”, CC shortens to “piece 3 is left of piece 1”, and so on.

In fact, there’s only 6 patterns of operators you ever need in S4; everything else will cancel down to one of these. These patterns are: M, C, MC, CM, MCM, CMC.

Now do exercise 5.

System S5: Could this ever be true?

System S5 is another way of interpreting what possibility and necessity mean. In S4, we looked at what happens in the future. If we started to ask claims about worlds mid-way down a modal diagram, it would still only look forwards. If we asked questions about the worlds at the bottom of each diagram, S4 would only look at that one situation. It looks forwards down each branch of the tree diagram, and won’t let you consider what might be happening in one of the other branches.

S5 works differently — as well as looking forwards to see what can happen, it also lets you look backwards, so that you can explore other sets of situations which might arise. Think of it as time travelling. If we stepped back in time, changed the way pieces moved, and then waited long enough, we could end up anywhere we wanted. What this means is that in S5, you can travel backwards and forwards along as many connections as you want; you can get from any one world to any other grid world in the diagram. Effectively, what this means is that each grid world is connected to every other world in the diagram.

This changes what stacked modal operators mean. Look again at the Modal diagram we’ve been using.

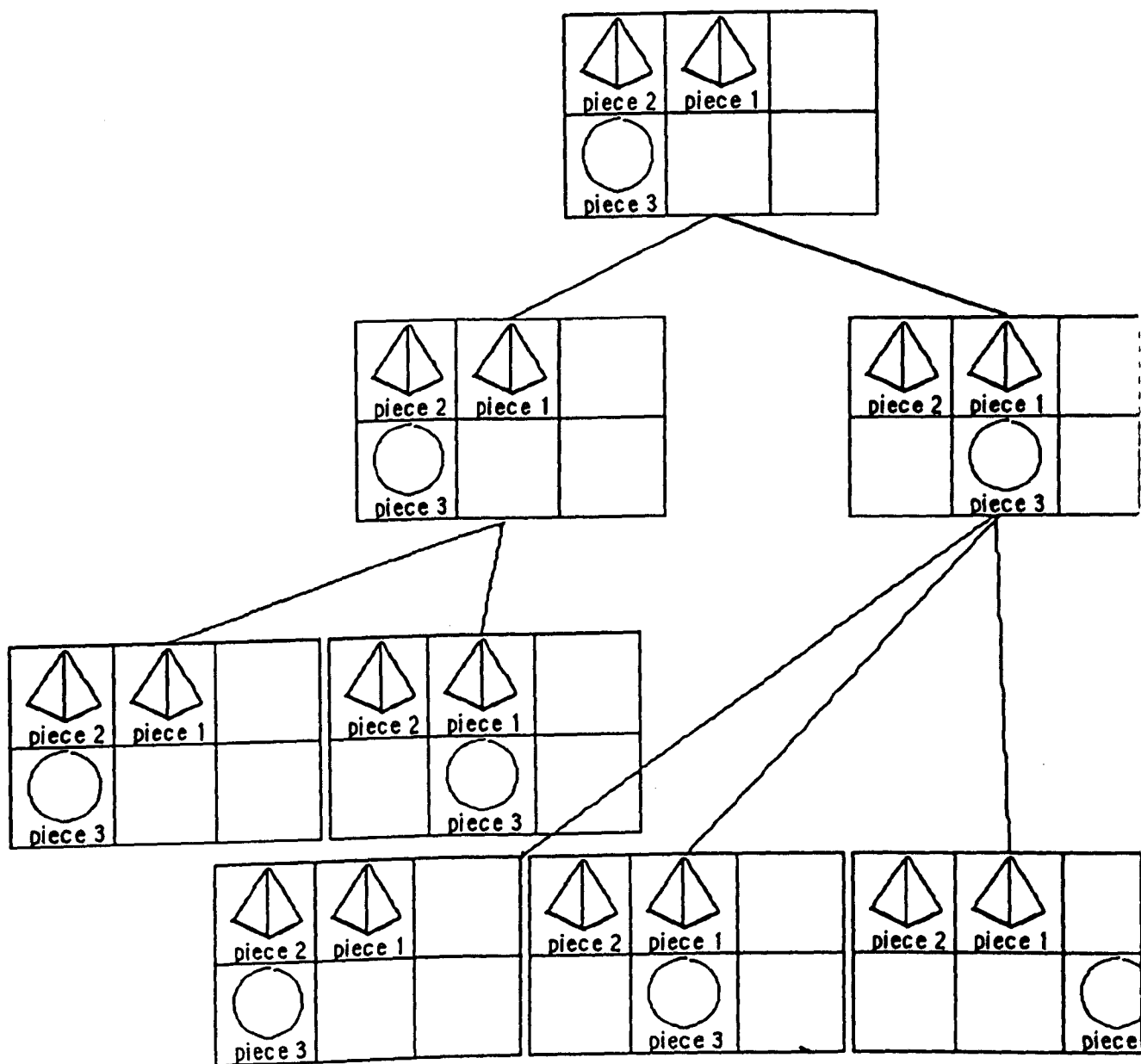


Figure 7: the extended modal diagram

The claim "piece 3 is left of piece 1" is true in at least one world. From any one world, you can get to all of the others; as a result, \Box "piece 3 is left of piece 1" will be true in every world.

Let's take another example. A completed modal diagram is shown below, in figure 8.

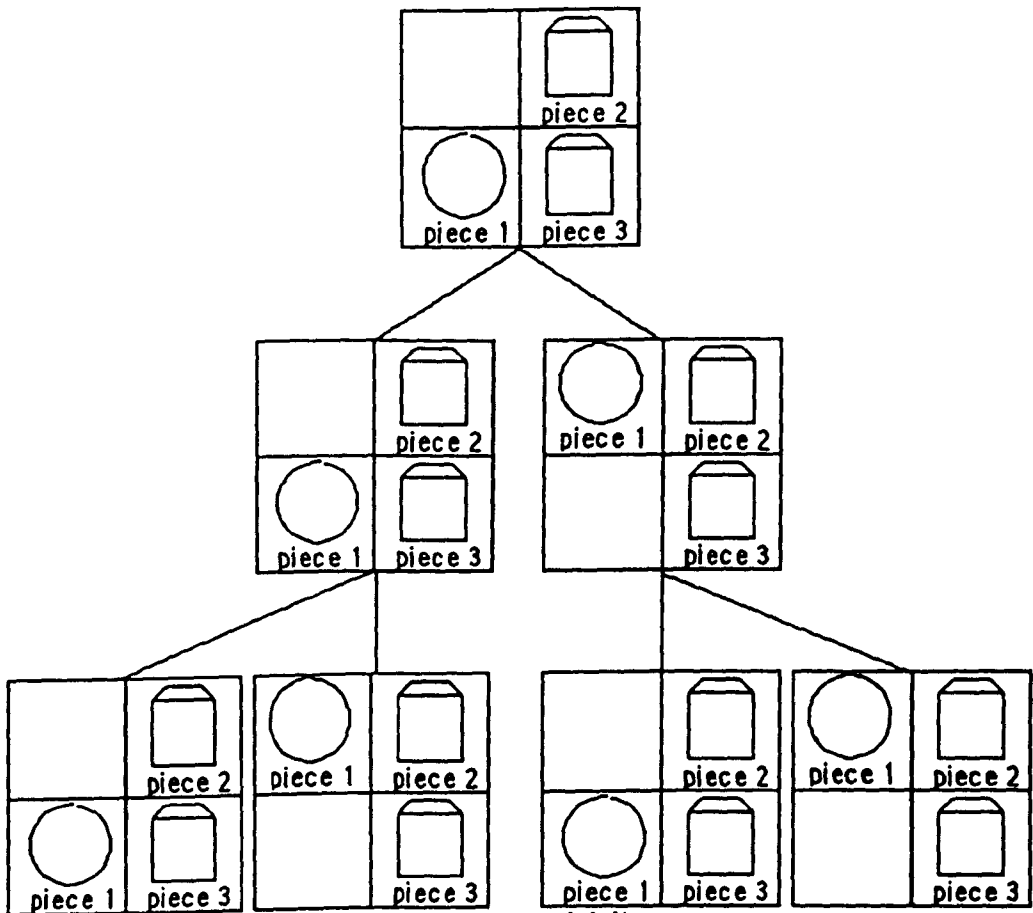


Figure 8: a new modal diagram

Here, M “piece 1 is left of piece 3” is true for our starting world. What this means is that it’s true in all the worlds you can get to from world 1. But in S5, you can move to *every* grid world, so “piece 1 is left of piece 3” must be true in all of the worlds in the diagram. Just as happened with the possibility claims (those starting with ‘C’), this means that if M “piece 1 is left of piece 3” is true in *any* world, it’s true in all of them.

What effect does this have? Well, if any modal claim is true, it’s always true, irrespective of what world you’re looking at. In other words, this modal claim *must* be true. (It’s also fair to say that the modal claim *can* be true, since it’s true for at least one world) In other words, if M “piece 1 is left of piece 3” is true, MM “piece 1 is left of piece 3” and CM “piece 1 is left of piece 3” are also true.

The effect of this is that you can simplify stacked modal operators. The only one you need to look at is the last one (the one closest to the statement) — in other words, you can get rid of all the modal operators except the last one. So, for example, CM “piece 1 is left of piece 3” simplifies to M “piece 1 is left of piece 3”. So will MCM “piece 1 is left of piece 3”, CCCM “piece 1 is left of piece 3”, MCMCM “piece 1 is left of piece 3”, and so will every other stack of Modal operators which ends in M “piece 1 is left of piece 3”.

Do Exercise 6. (You have now completed the instructional material. If you want, you can spend any remaining time creating your own examples.)

Exercises

N.B. Please tick each question you attempt!

(Answers can be given on separate sheets of paper)

Exercise 1

- i) Draw (or create, using the software) a grid world with a 3x3 grid, with one circle in the top right hand corner (called piece 1), one tetrahedron in the bottom left (called piece 2), and a cube in the centre (called piece 3).

Exercise 2

- i) Draw a modal diagram for the world you created in exercise 1. Check that three new worlds are created. Why is this?

Exercise 3

- i) For your diagram, see whether the claim M "piece 1 is right of piece 2" is valid.
- ii) Is M "piece 1 is above piece 3" valid?

Exercise 4

- i) For the diagram you have created, show that MM "piece 1 is right of piece 2" is false.
- ii) Is CC "piece 1 is right of piece 2" true?
- iii) Is MC "piece 1 is right of piece 2" true?

Exercise 5

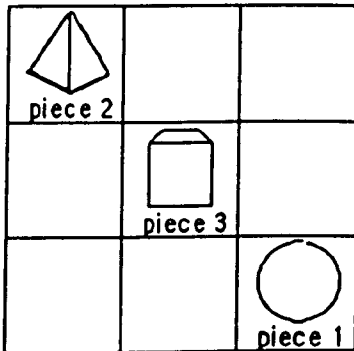
(If you are using the computer, use the "Set up initial conditions" button to switch to system S4)

- i) Keeping the diagrams you used in the last exercise, show that in S4, if CC "piece 3 is left of piece 1" is true, so is C "piece 3 is left of piece 1".
- ii) Show that MM "piece 3 is left of piece 1" being false means that M "piece 3 is left of piece 1" is false, too.
- iii) Does MMC "piece 1 is right of piece 2" being true mean that MC "piece 1 is right of piece 2" is true, too?

Exercise 6

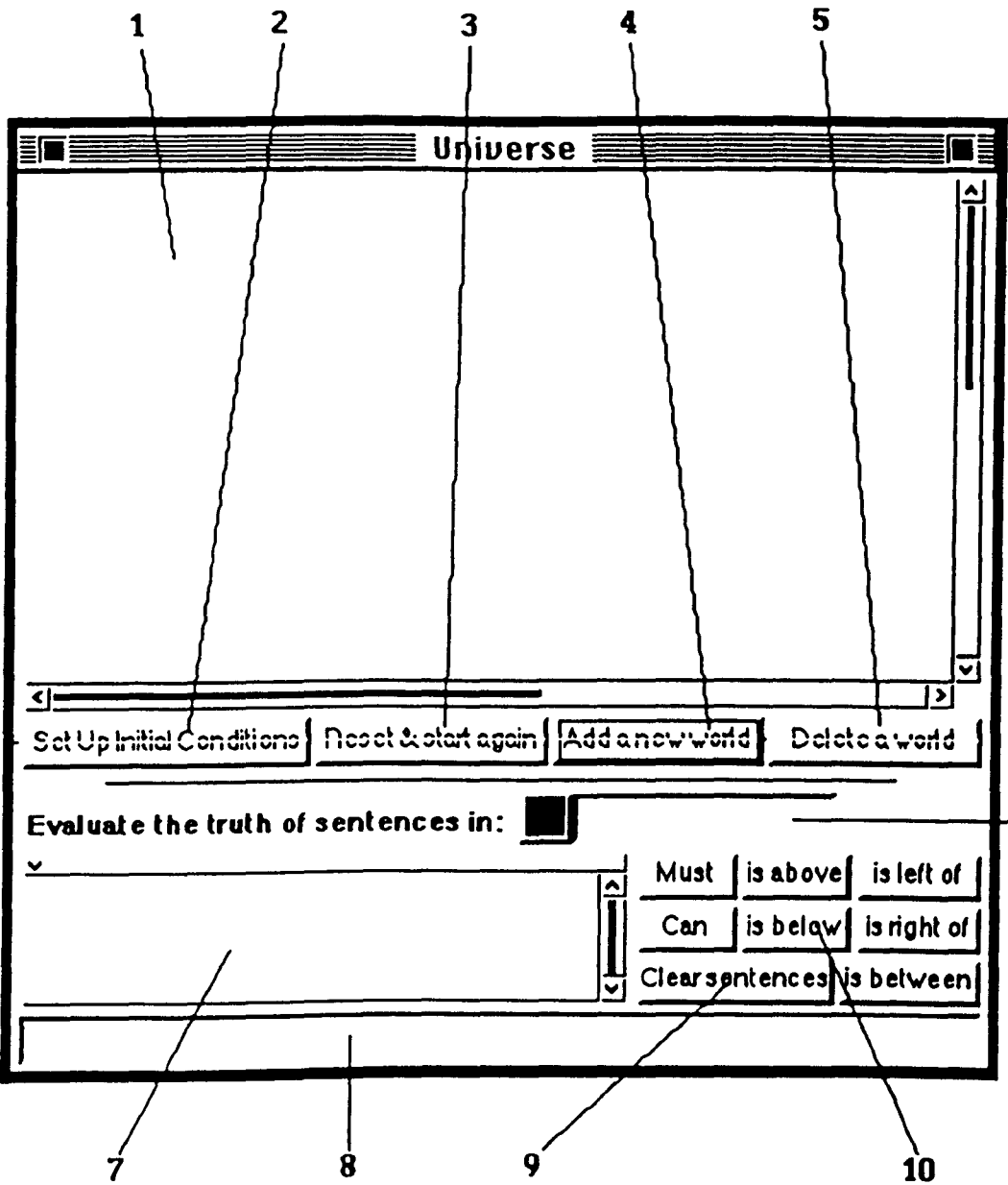
(If you are using the computer, use the "Set up initial conditions" button to switch to system S5)

- i) Construct a diagram, starting from the world shown below. Let pieces move twice.



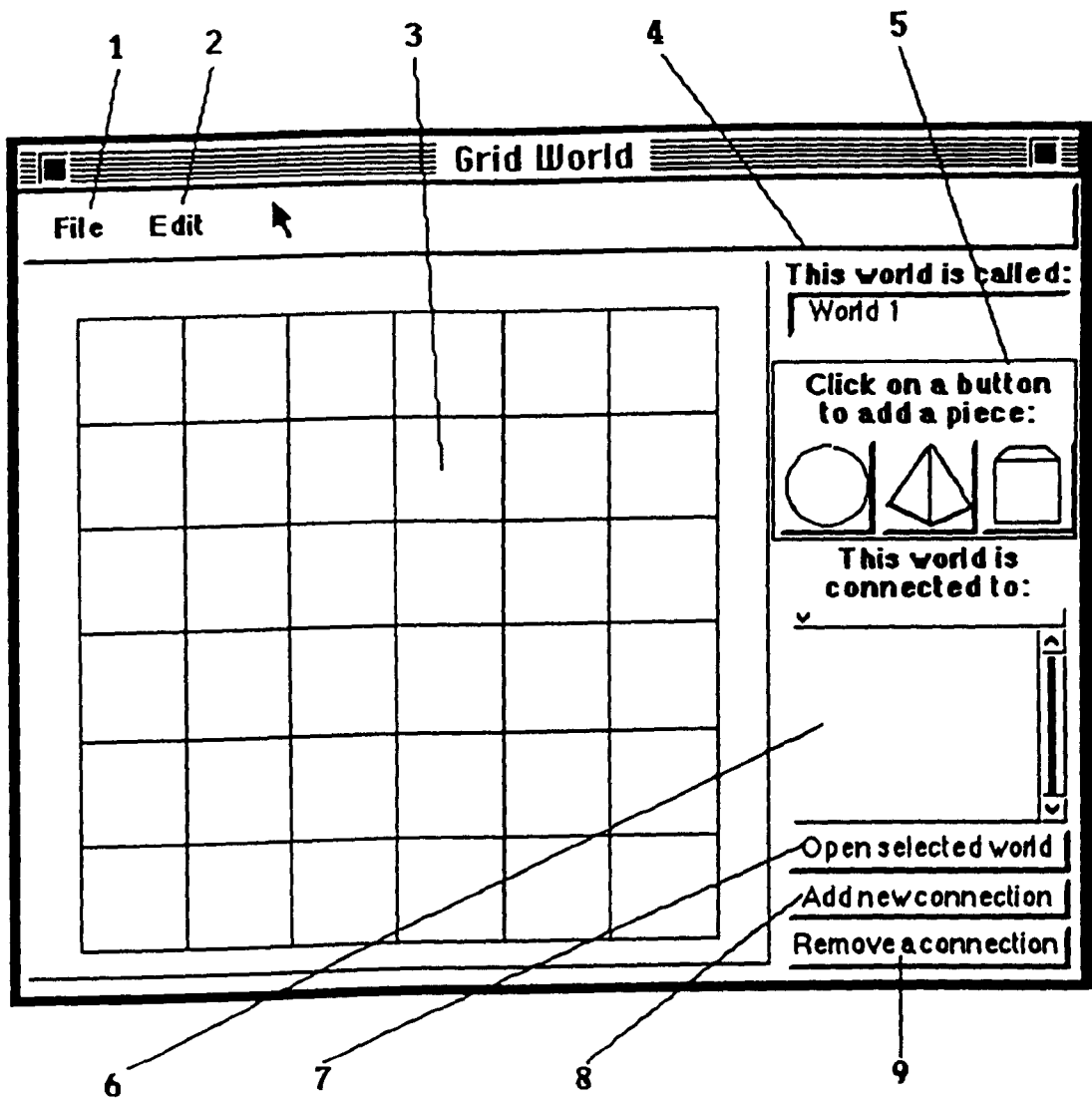
- ii) Show that the claim M "piece 1 is below piece 2" is false in every world.
- iii) Show that C "piece 1 is left of piece 3" is true in every world.
- iii) Say whether CM "piece 1 is below piece 2" is true.
- iv) Is MC "piece 1 is below piece 2" true?
- v) Is CMC "piece 1 is below piece 2" true?
- vi) Is MMM "piece 3 is below piece 2" true?

Software prompt sheet



- (1) **Universe window.** This will keep track of all the grid worlds you create, and how they are connected. You can move worlds by dragging and dropping them, and can double-click worlds to open them.
- (2) **Set-up button.** Use this to do things like alter the size of the grids in the grid worlds, change the modal system you're using, and allow pieces other than spheres to move.
- (3) **Reset and start again.** This button gets rid of all the worlds you've created.
- (4) **Add a new world.** Pressing this button won't seem to do much. What you need to do is press the button, then move the cursor into the universe window, where it will change into a crosshair, to show you're about to place a world. Clicking inside the window will place a new world at the cursor point.
- (5) **Delete a world.** This works in a similar way to the "Add a world" button. Press it once, move it into the universe window, and then click on the world you want to delete. When you are about to delete a world, the cursor changes into a circle with a bar through it.
- (6) **"Evaluate truth of sentences in world: "** Pressing the button to the right of this sentence offers you a choice of all the worlds currently created. Select one of these in order to start evaluating sentences in that world. The text box right of the button will display the name of the world you have chosen.
- (7) **The sentence window.** When you create sentences, they will be displayed here.
- (8) **The shorthand code window.** Since sentences can be hard to follow, this text window displays a shorthand version of the sentence.
- (9) **Clear sentences button.** This wipes the sentence window and the shorthand code windows clean.
- (10) **Sentence creation buttons.** Use these to build up the sentences you want to evaluate. Any of these can be used to start a sentence, but only the comparison buttons (is above, is below, is left of, is right of, is between) will complete a sentence. When you choose one of these, a dialogue window will open up, and ask you to select which pieces you want to compare.

You can add as many "must" and "can" sections to the sentence as you want, by pressing the must and can buttons repeatedly.



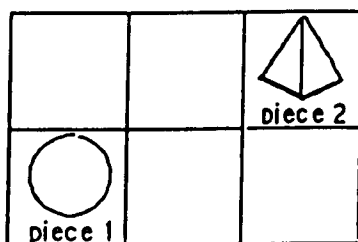
- (1) **File menu.** You can use this to close the window.
- (2) **Edit menu.** This offers you a selection of commands which can be used to edit the contents of the grid world. The command you are most likely to need is, **"move pieces and create new worlds"**.
- (3) **Grid window.** This displays the grid, and whatever pieces you have placed so far. Pieces can be dragged and dropped, and double-clicking on them allows you to edit or delete them.
- (4) **"This world is called: "** This window shows the name of the grid world. You can edit this if you wish.
- (5) **"Press any button to add a piece"** These three buttons work in the same way as the **"add a world"** button (4) in the universe window. Click (and release) whichever shaped piece you want to add, and then click inside the grid to place the piece.
- (6) **"This world is connected to: "** This window gives a list of all the other grid worlds this one is connected to.
- (7) **Open selected world.** When a connection is highlighted in the list above the button, clicking here will open a new window on that grid world.
- (8) **Add a connection.** Clicking this button gives a list of all the grid worlds you have created so far. Choosing one will add it to the list of the world's connections.
- (9) **Remove a connection.** When a connection in the list is highlighted, pressing this button will remove the connection.

Post-Test: Please attempt any questions you think you might be able to answer. Use extra sheets of paper if you need to, and hand them in with this test.

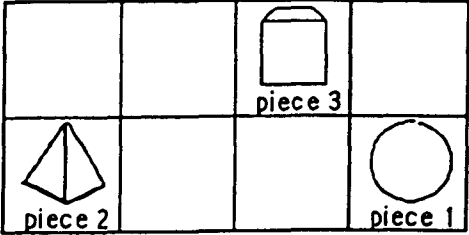
- 1) Give a brief explanation of what you think "accessibility relations" are.

- 2) Briefly explain the differences between the Modal systems T, S4 and S5.

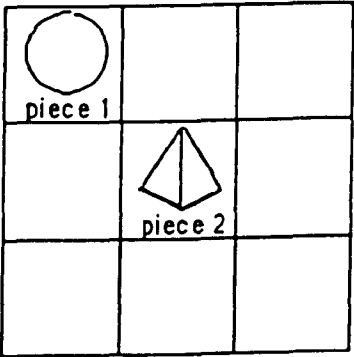
- 3) Given the following diagram, prove (in system T) that it must be the case that piece 1 is left of piece 2. Piece 1 is allowed to move up to one square left, right, up or down. "piece 1 is left of piece 2" should be understood to mean that piece 1 is in a column which is left of the one containing piece 2.



- 4) Given the following diagram, show that the claim, "It can be the case that piece 1 is between piece 2 and piece 3" can be true in S4 and S5, but not in T, given the accessibility relation that piece 1 can move up, down, left or right by up to one square in any one movement. "Piece 1 is between piece 2 and piece 3" should be taken to mean either that piece 1 is in a column left of one piece and right of the other, or that it is in a row above one piece and below the other. (Hint: the accessibility relation can be applied more than once)

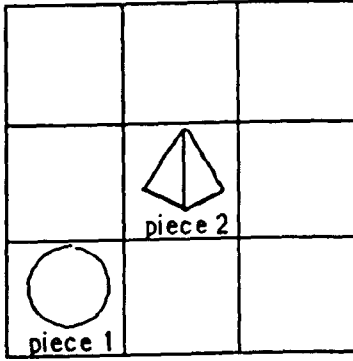


- 5) For each of the following claims, state in which Modal systems (T, S4, S5) they will be valid. (N.B. M is shorthand for, "it must be the case that...", C is shorthand for, "it can be the case that...", and the symbol " \rightarrow " means "implies")
- a) CM "piece 1 is above piece 2" is true if CMCCMM "piece 1 is above piece 2", for the situation:



(i.e. that CMCCMM "piece 1 is above piece 2" \wedge CM "piece 1 is above piece 2")

- b) C "piece 1 is above piece 2" is true if MCMCCC "piece 1 is above piece 2", for the situation:



(i.e. that MCMCCC "piece 1 is above piece 2" \rightarrow C "piece 1 is above piece 2")

- c) MCMp is true if MCMMMCp is true, where p is shorthand for any true claim.
(i.e. MCMMMCp \rightarrow MCMp)

- d) CMp is true if MCMCMCp is true, where p is shorthand for any true claim.
(i.e. MCMCMCp \rightarrow CMp)

- 6) Let U be a new modal system, in which situations never change. In other words, when evaluating claims, you can only consider whether or not the claim is true in your starting world. Put another way, the system's accessibility relation is an identity relation.

Give an example of when such a system could occur. Show that, for a claim p which is true in world 1 (your starting world), the claim "It must be the case that p" is true. (In other words, show that $p \rightarrow Mp$)

Post-Test: Part B

- 1) Did you enjoy taking part in this study?
Yes..... ☐
It was Ok..... ☐
No..... ☐

- 2) How well do you think you did?
Well — got quite a few post-test answers right ☐
Ok — did a little bit better on the post-test than the pre-test..... ☐
Badly — didn't improve at all, or got worse ☐

- 3) Did you have any problems constructing Modal proofs in the post-test? (e.g. questions 3 & 4)
No, not really ☐
Yes..... ☐

(If you had problems, could you say briefly what they were:
.....
.....
.....
.....)

- 4) Did you understand system T?
Not at all..... ☐
I could use it, but didn't really understand it..... ☐
I understood it pretty well..... ☐

(If you didn't understand it, please say what was difficult about it:.....
.....
.....
.....)

5) Did you understand system S4?

- Not at all..... ☐
- I could use it, but didn't really understand it..... ☐
- I understood it pretty well..... ☐

(If you didn't understand it, please say what was difficult about it:
.....
.....)

6) Did you understand system S5?

- Not at all..... ☐
- I could use it, but didn't really understand it..... ☐
- I understood it pretty well..... ☐

(If you didn't understand it, please say what was difficult about it:
.....
.....)

7) Did you find the grid world examples easy to understand?

- Yes..... ☐
- No..... ☐

(If no, please say why:
.....
.....)

8) What did you think about the number of examples in the teaching material?

- Too many..... ☐
- About right..... ☐
- Too few..... ☐

- 9) Was there anything particularly good about using pen & paper to work through the teaching material?
- 10) Was there anything particularly difficult about using pen & paper to work through the teaching material?
- 11) If you had used a piece of software which had helped you draw out the proofs, do you think you would have worked through more examples?
Yes..... ☐
No..... ☐
- 12) Would you have preferred to use pen & paper or software when working through the teaching material?
Pen & paper ☐
Software..... ☐

Please say why:.....
.....
.....
.....
.....

- 9) Was there anything particularly good about using the computer to work through the teaching material?
- 10) Was there anything particularly difficult about using the computer to work through the teaching material?
- 11) If you had used pen & paper, which would have given you more control over how you constructed the diagrams, do you think you would have worked through more examples?
Yes..... ☐
No..... ☐
- 12) Would you have preferred to use pen & paper or software when working through the teaching material?
Pen & paper ☐
Software..... ☐

Please say why:.....
.....
.....
.....
.....

Appendix 9: Quantitative data, third study

This appendix contains quantitative data from the study described in chapter 6. These data are coded as follows:

Condition:	1 denotes pen & paper, 2 denotes software
Gender:	1 denotes male, 2 denotes female
Prior logic:	The subjects' previous experience with logic, with 1 as none at all, 2 as currently taking a first course in logic, 3 as having completed a first course in logic, and 4 as having completed a course in logic and studied a type of non-standard logic (e.g. fuzzy logic, Modal logic, etc).
Exercises:	The number of questions the subject attempted whilst working through the instructional material, ranging from 0 to 17.
Pre-test AR:	Denotes whether the subject failed to attempt this question (0), offered a correct explanation of what accessibility relations are (1), or gave an incorrect explanation (2).
Post-test AR:	As above.
T/S4/S5:	Scored from 0 to 3, according to how many of the Modal systems were correctly defined in question 2.
Proof Construction:	Rated as 0 if the subject did not attempt to construct a proof, 1 if the subject managed to provide a valid proof, and 2 if the subject provided an incorrect proof.
Examples:	Rated at 1 if the subject used either examples or counterexamples as part of the proof process in the tests, and at 2 if they did not.
Reduction:	This records the number of systems correctly given or omitted on the stacked Modal operator cancellation questions, ranging from 0 to 12.
Reduction type:	This records which type of reduction problems the subject attempted. 0 indicates that no questions were attempted, 1 that only concrete questions (those with a grid world example) were attempted, and 2 that both concrete and abstract questions were attempted.
Near Transfer:	Subjects who made no attempt to answer this question are denoted by a 0, those who successfully completed the question are denoted by a 1, and those who attempted the question but failed to complete it successfully are denoted by a 2.
Understood T:	Subjects' own assessment of their understanding of system T. 1 denotes no understanding, 2 indicates that the subject can apply the system, but does not really understand it, and 3 that the subject understands the system.
Understood S4:	As above, but for S4.
Understood S5:	As above, but for S5.
Understood Proof:	Subjects' self-assessed ability to construct proofs. 1 denotes that the subject considered themselves able to construct proofs, and 2 that they considered proof construction to be problematic.
Enjoyment:	Subjects' self-assessed enjoyment of taking part in the study. Rated from 1 (didn't enjoy it), through 2 (it was o.k.) to 3 (enjoyed it).
Confidence:	Subjects' self-assessed performance on the post-test. Rated at 1 if the subject felt they had done badly, 1 if they thought they had done 'o.k.', and 2 if they believed they had done well.
Understand grids:	Rated at 1 if the subject found grid world examples easy to understand, and 2 if they did not.
Enough exercises?:	Recorded whether the subject felt there were too many (1), too few (3), or about enough (2) exercises in the teaching material.
Pros:	Subjects who gave an example of something particularly good about their condition are denoted by a 1; those who did not give an example are denoted by a 2.
Cons:	Subjects who gave an example of something particularly bad about their condition are denoted by a 1; those who did not give an example are denoted by a 2.

Prefer:

1 denotes that the subjects would have preferred to use pen and paper to complete the teaching material, and 2 denotes that they preferred MoLE.

Subject	Condition	Gender	Prior logic	Exercises	Pre-test AR	Post-test AR
1	1	1	1	10	0	1
2	1	1	1	7	0	1
3	1	1	1	7	1	1
4	1	1	1	9	0	1
5	1	1	1	7	0	2
6	1	1	1	10	0	1
7	1	1	2	4	0	1
8	1	1	3	7	0	2
9	1	1	3	7	0	1
10	1	1	3	7	0	2
11	1	1	3	7	0	2
12	1	1	3	4	0	2
13	1	1	3	4	0	0
14	1	1	3	7	0	1
15	1	1	3	13	0	1
16	1	1	3	10	0	1
17	1	1	4	7	0	1
18	1	2	1	7	0	2
19	1	2	1	7	0	1
20	1	2	2	10	1	1
21	1	2	3	4	0	2
22	1	2	3	7	0	1
23	2	1	1	7	0	1
24	2	1	1	7	0	1
25	2	1	1	10	0	1
26	2	1	1	17	0	2
27	2	1	1	17	0	1
28	2	1	1	2	0	1
29	2	1	3	17	0	1
30	2	1	3	7	0	1
31	2	1	3	9	0	1
32	2	1	3	15	0	1
33	2	1	3	17	0	0
34	2	1	4	15	0	1
35	2	2	1	10	0	2
36	2	2	1	7	0	1
37	2	2	3	7	0	2
38	2	2	3	7	0	0
39	2	2	3	4	0	1
40	2	2	4	7	0	1

Subject	Condition	T/S4/S5	Proof Construction	Examples	Reduction	Reduction type	Near transfer
1	1	0	2	2	0	0	0
2	1	0	2	2	0	0	0
3	1	2	2	2	0	1	0
4	1	3	1	1	1	1	1
5	1	3	1	1	0	0	0
6	1	0	1	2	2	1	0
7	1	2	1	1	9	2	1
8	1	2	1	2	0	0	0
9	1	3	1	1	9	2	1
10	1	3	1	2	2	1	0
11	1	3	2	1	3	1	0
12	1	0	2	2	0	0	0
13	1	0	2	2	0	0	0
14	1	2	1	1	3	2	1
15	1	0	1	1	0	0	1
16	1	3	1	1	5	2	0
17	1	1	2	1	0	0	0
18	1	3	1	1	0	0	0
19	1	2	1	0	0	0	0
20	1	2	1	2	6	2	1
21	1	0	2	0	0	0	0
22	1	0	2	2	0	2	0
23	2	3	1	1	4	1	1
24	2	3	1	1	5	1	0
25	2	3	1	2	8	2	0
26	2	3	1	1	10	2	0
27	2	3	1	2	10	2	1
28	2	0	2	2	0	0	0
29	2	3	1	1	10	2	1
30	2	3	2	1	4	2	1
31	2	3	1	1	11	2	1
32	2	3	1	1	9	2	1
33	2	0	1	1	2	1	0
34	2	3	1	1	10	2	1
35	2	2	2	1	0	0	2
36	2	2	1	1	2	1	0
37	2	1	1	2	0	0	0
38	2	0	2	2	0	0	0
39	2	3	1	2	7	2	2
40	2	3	1	1	10	2	0

Subject	Condition	Understand T	Understand S4	Understand S5	Understand Proofs	Enjoyment	Confidence
1	1	3	2	2	2	2	2
2	1	2	2	2	1	2	2
3	1	3	2	1	1	2	2
4	1	3	3	2	1	3	3
5	1	2	2	2	2	3	2
6	1	3	3	2	2	2	2
7	1	3	3	1	2	3	3
8	1	3	2	1	2	3	2
9	1	3	3	1	2	3	2
10	1	3	2	2	1	2	2
11	1	2	2	2	1	3	2
12	1	1	2	1	1	3	1
13	1	2	1	1	2	2	1
14	1	3	3	1	1	3	2
15	1	3	3	3	1	3	2
16	1	3	3	2	2	3	3
17	1	3	2	3	2	3	2
18	1	3	2	1	2	2	2
19	1	3	3	3	1	3	3
20	1	2	2	1	2	2	2
21	1	1	1	1	2	1	1
22	1	2	2	1	2	3	2
23	2	3	3	3	1	2	3
24	2	2	3	1	1	3	3
25	2	3	2	2	2	3	2
26	2	3	3	2	1	2	3
27	2	3	3	3	2	2	3
28	2	1	1	1	2	2	1
29	2	2	2	3	1	2	2
30	2	3	3	3	1	3	3
31	2	3	3	3	1	3	3
32	2	3	3	2	1	3	3
33	2	3	2	2	2	3	2
34	2	3	3	2	1	3	3
35	2	2	2	1	1	2	2
36	2	3	2	1	1	2	2
37	2	3	1	1	1	2	2
38	2	3	3	3	2	3	2
39	2	3	3	1	1	2	3
40	2	3	3	3	1	3	3

Subject	Condition	Understand grids	Enough exercises?	Pros	Cons	Prefer
1	1	1	1	2	1	2
2	1	1	2	2	1	2
3	1	1	2	2	2	2
4	1	1	2	2	1	2
5	1	1	3	1	1	2
6	1	1	2	1	2	1
7	1	1	2	2	1	2
8	1	1	1	2	1	2
9	1	1	2	2	1	2
10	1	1	2	1	1	2
11	1	1	2	1	2	1
12	1	1	2	2	2	2
13	1	1	2	1	1	1
14	1	1	2	2	1	2
15	1	1	2	2	2	1
16	1	1	2	1	1	2
17	1	1	2	1	2	2
18	1	1	2	1	2	2
19	1	1	2	1	2	2
20	1	1	2	2	1	2
21	1	1	3	2	1	2
22	1	1	3	2	2	2
23	2	1	2	1	2	1
24	2	1	2	1	2	2
25	2	2	2	1	1	2
26	2	1	3	1	2	2
27	2	1	2	1	2	2
28	2	2	2	1	2	2
29	2	1	2	1	2	2
30	2	1	1	1	2	2
31	2	1	2	1	1	2
32	2	1	2	1	2	2
33	2	1	2	1	1	1
34	2	1	2	1	1	2
35	2	2	3	2	1	1
36	2	1	2	1	2	2
37	2	1	2	1	2	2
38	2	1	2	1	2	2
39	2	1	3	2	2	1
40	2	1	2	1	2	2

Appendix 10: Qualitative data, third study

This appendix contains subjects' responses to open questions on the post-test. Subjects have been ordered by condition, gender and expertise (with 'experts' having completed a first course in logic, and 'novices' not having done so). Within each subject, comments have been coded to indicate the topic they refer to.

- Proof: Responses to the question asking if subjects had encountered difficulties constructing Modal proofs.
T: Responses indicating any problems the subject had with system T
S4: As above, for system S4.
S5: As above, for system S5.
Grids: Problems subjects had in understanding grid world examples.
Pros: Advantages to the condition the subject had been assigned to.
Cons: Disadvantages to the condition the subject had been assigned to.
Prefer: When asked for a preference for using pen & paper or MoLE, subjects were asked to explain why they had made this choice.

Pen & Paper: Male novices

Subject 1

- Proof: Understanding and taking everything in.
S4: Just complicated. Needed more time to figure it out.
S5: Understood concept, just couldn't grasp it.
Cons: Slow. Too many crossings out.
Prefer: Change mistakes easily and faster. Software would also give basic guidelines so you get a better idea of what's expected from you.

Subject 2

- T: As for the other two systems they were a bit confusing.
Cons: Drawing the grid worlds.
Prefer: Some times it is easier if you can work with software. The visual representation of it can help with the solving of problems.

Subject 3

- S5: Didn't read it before the post-test.
Prefer: Computer would eliminate time-consuming drawing of the worlds.

Subject 4

- S5: I didn't reach that part in the notes, though I think I understood the principle from the earlier sentence.
Cons: Drawing grids & symbols (not exactly my strong point).
Prefer: It would save time with the hand drawings, i.e. slightly easier.

Subject 5

- Proof: Not sure of the order of evaluation of CM.
Pros: Having the teaching material there in front of me, and being able to carry it with me.
Cons: Spend too much time on drawing grid worlds and symbols.
Prefer: Easier to draw grids and symbols, and have better overview, especially in more steps in advance.

Subject 6

- Proof: Question 5 was confusing; did not understand it at all.
S5: Could not follow M implies MM etc.
Pros: Could go back and review previous work, and correct mistakes if I noticed them later.
Prefer: Reasons given in 9. Easy to review previous work and correct errors.

Subject 7

- Proof: Didn't get time to study S5.
Cons: Takes a long time to generate diagrams.
Prefer: If the examples can be done more quickly, then more time can be spent learning new material.

Pen & Paper: Male experts

Subject 8

- Proof: I didn't grasp how to apply S4 and S5. I was ok with T, a bit confused with S4, and lost with S5.
S4: I understood that loops were generated, but didn't know how to apply this to be able to sort out the problems.
S5: It was the last bit of theory, and after having struggled unsuccessfully with S4, I wasn't ready for S5. I would have needed time to rest.
Pros: No, it makes you write several times redundant grids.
Cons: Yes, you waste time copying previous exercises, to do some extra work on them.
Prefer: It would have been less messy and more efficient.

Subject 9

- Proof: Wasn't sure when to draw pictures or up to which point drawing consisted of a proof.
Cons: Takes a lot of time.
Prefer: I suppose it would have been easier.

Subject 10

- S4: Need longer to familiarise myself with it as it is a bit more abstract than T.
S5: Even more abstract than S4, need to be familiar with it.
Pros: Find I take more in by using pen & paper, think about it more.
Cons: Yes, drawing the modal diagrams could be a bit tedious after two or three layers.
Prefer: Would have spent less time drawing diagrams, and therefore have more time to spend answering questions and learning the concepts. Useful to (pen & paper) draw small diagrams so (to start with) as to be sure what's happening, without computer assistance.

Subject 11

- T: I had only just read about it and forgot half of it, but with more examples to work through, I would be able to understand it.
Pros: You could cross parts out easier, it could have maybe been faster than using software but I am not sure.
Cons: It looked messy (well, for me, anyway!)
Prefer: I could take more notes as I went along, rather than only being able to do examples using the software.

Subject 12

- T: Firstly, I never used system T in logic, therefore I did not know from where to start, especially if you have a limited given time.
S5: The time did not allow me to reach the final question.
Cons: Because you are not sure about the object you have drawn, whether it is true or false.
Prefer: Maybe the pictures would have been clearer.

Subject 13

- Pros: For me, writing on paper is good because it helps me concentrate more.
Prefer: If I use pen & paper, I would spend more time trying to grasp the material.

Subject 14

- S5: Didn't have enough time
Cons: Time consuming, maybe.
Prefer: Seems less like work, and you don't have to do all of the tedious stuff.

Subject 15

[No comments given]

Subject 16

- Proof: High complexity of trying to conceptualise some of the accessibility relations (e.g. MCMCMCM etc).
- S4: More thought needed - could be a bit tricky.
- S5: Seemed a bit "abstract" although I thought I was beginning to understand it.
- Grids: Grid worlds seemed like a good method to convey the ideas. Simplistic and easy to understand.
- Pros: Could easily write down/draw ideas.
- Cons: Can be a bit hectic!
- Prefer: I did not use software, but pen and paper seemed adequate.

Subject 17

- Proof: Remembering what the difference between S4 and S5 is.
- S4: It seems a bit vague.
- Pros: Yes, it encourages you to take notes.
- Prefer: It guides you better, i.e. stops you doing things too soon.

Pen & Paper: Female novices

Subject 18

- Proof: I didn't know what the exact form of the proof should be like.
- S4: Didn't read enough about it.
- S5: Had no time left to read about it.
- Pros: Making fast scratches.
- Prefer: It might help me to understand things faster.

Subject 19

- Pros: I liked the use of example diagrams, just being able to flick back and forth.
- Cons: Looking through the text for reference - but the bold text made that easier.
- Prefer: It would have helped me draw out the proofs, and work through more examples.

Subject 20

- Proof: Forgot how to do it.
- S4: Didn't understand T.
- S5: Didn't understand T or S4.
- Cons: Drawing diagrams neatly.
- Prefer: Faster to draw diagrams.

Pen & Paper: Female experts

Subject 21

- Proof: I think the text was clear, but there needed to be more simple examples to see, and also I needed time to go over the material properly.
- T: I don't feel I had enough time to go through everything.
- S4: I didn't even get to this point.
- Cons: You had to keep referring back to past papers and the text a lot.
- Prefer: I'm assuming that there would be on-line help, and something to tell you when and why you are making errors in constructing the proofs and this would obviously make it easier to learn and also be more useful than referring to text and writing it all down on paper. I felt because I was constrained just to the text for help I was very unguided, because sometimes it was hard to follow the text, which made matters worse for me!

Subject 22

- Proof: I didn't quite grasp the difference between T, S4 and S5.
- S5: I didn't get that far in the exercise.

Computer: Male novices

Subject 23

- Proof: The software was a little confusing compared to drawing the diagrams by hand.
Pros: Diagrams were easy to draw.
Cons: Learning exactly what it was producing in a short space of time.
Prefer: Initially, I would have preferred pen and paper, but I think with practice the software would have become easier.

Subject 24

- T: I'm still a bit unsure of the worlds it caters for - is it just world 1 or the next level of worlds or both?
S5: Didn't have time to read about it.
Pros: It was easier than trying to work things out on pen & paper.
Prefer: Software is just easier, especially if someone has little or no knowledge of logic. On pen & paper, mistakes can be made without realising it, but the computer should highlight problems.

Subject 25

- Proof: Wasn't sure why I knew that MCMCMCCM=MCM or whatever.
Grids: Extensions were a little complex, as were repeats.
Pros: Faster at drawing diagrams i.e. less tedious.
Cons: Still takes time.
Prefer: Easier, neater, less changeable.

Subject 26

- Pros: It was easier to test things in different worlds.
Prefer: Drawing the diagrams would have been tedious.

Subject 27

- Proof: I am not sure how to construct the proof.
Pros: It would have been a pain to draw the worlds.
Prefer: Easier than drawing them out; did not have to look at diagrams to see when true or false.

Subject 28

- Proof: I was not able to complete the task.
T: Could not complete this task in time.
Grids: I started the exercise with difficulty because I couldn't understand the terminology in the text. My lack of knowledge in this area, I think. However, I went through the text again using pen and paper to understand the principle behind it. This time I managed to get as far as exercise 2. Given time, I may have been able to complete this task, but I think I will need more time.
Pros: To draw possible worlds etc.

Computer: Male experts

Subject 29

- Pros: It demonstrated the material in a way that was easy to see and understand without wasting time diagram drawing.
Prefer: It was much faster and convenient, which means the work can be completed faster.

Subject 30

- Grids: Very easy.
Pros: Live examples.
Prefer: It's better to understand, quicker and you just "catch" the example, not need to think how to construct examples.

Subject 31

- Proof: The computer was too slow, and the diagrams messy.
- Grids: As above, computer presentation problems, messy, too slow.
- Pros: It was faster and drew diagrams, which saved time, reduces errors. However, if the user didn't understand the method of a relation this could be ignored which is inadvisable.
- Cons: It was slow, and the diagrams of the world were messy which was a major problem. Otherwise very good.
- Prefer: Drawing diagrams by hand is tedious & prone to random errors. This sort of backtracking/extension is suitable to computers not humans. However, one should be required to draw diagrams occasionally to show one has a correct and suitable understanding of the system.

Subject 32

- Pros: Yes, you don't have to draw lots of diagrams, which are messy and make it more difficult to understand the essential principles.
- Prefer: Makes it easier sometimes, you can concentrate on actual material rather than messy diagrams. On the other hand, it is sometimes easier to do it by hand and it can make you understand things better.

Subject 33

- Proof: Understood it vaguely but not entirely sure. So made the answering questions difficult.
- S4: Just too short for all these new ideas in 45 minutes.
- Pros: It saves a lot of time computing for all the worlds, and could understand more from the examples.
- Cons: When you don't understand it properly before, you'll get any benefit from it as pressing button on the computer doesn't guarantee you know what you are doing.
- Prefer: You could know more this thing by working through it (if one is willing to). But it is difficult and time consuming. I think pen and paper first, and then see the result in the software is the best.

Subject 34

- Pros: Saved time drawing diagrams.
- Cons: Not easy to extend diagrams - had to open lots of worlds.

Computer: Female novices

Subject 35

- S5: I just didn't really grasp the concept in the time allowed.
- Grids: I found it much easier using my own diagrams than trying to battle with new software - though I suppose that was just a case of my not being used to it.
- Cons: I'm not particularly good with new software anyway, and I just didn't feel the need for computer graphics for this work.
- Prefer: Easier to control, and also when you do it for yourself on pen & paper, you're made to actually consider the problem properly.

Subject 36

- S5: Didn't get that far.
- Pros: Easy to draw diagrams, so you can concentrate on ideas.

Computer: Female experts

Subject 37

- Pros: Easy to construct the diagrams and view the diagrams to solve the problems.
- Cons: Initially I was unsure; however, on reading the teaching material it was quite clear.
- Prefer: Faster, easier and more efficient. Clearer to view diagrams.

Subject 38

- Proof: I spent the time playing with the software without grasping the specific difference between the three systems. This is a fine case of RTFM property!
- Pros: The interaction imprints information on the memory.
- Prefer: It's much easier to see where you are when you can layer things. Working through again with pen and paper would probably consolidate.

Subject 39

- S5: I could not understand well the difference between S4 and S5.
- Prefer: I am used to using pen and paper every time I go through teaching material.

Subject 40

- Pros: Yes, it made it easier to understand because of the graphical representation.